



Titre: Ré-optimisation d'horaires de personnel en ajoutant des transferts
Title: entre départements

Auteur: Sarra Souissi
Author:

Date: 2016

Type: Mémoire ou thèse / Dissertation or Thesis

Référence: Souissi, S. (2016). Ré-optimisation d'horaires de personnel en ajoutant des transferts entre départements [Master's thesis, École Polytechnique de Montréal].
Citation: PolyPublie. <https://publications.polymtl.ca/2397/>

 **Document en libre accès dans PolyPublie**
Open Access document in PolyPublie

URL de PolyPublie: <https://publications.polymtl.ca/2397/>
PolyPublie URL:

Directeurs de recherche: Guy Desaulniers, & François Soumis
Advisors:

Programme: Maîtrise recherche en mathématiques appliquées
Program:

UNIVERSITÉ DE MONTRÉAL

RÉ-OPTIMISATION D'HORAIRES DE PERSONNEL EN AJOUTANT DES
TRANSFERTS ENTRE DÉPARTEMENTS

SARRA SOUISSI
DÉPARTEMENT DE MATHÉMATIQUES ET DE GÉNIE INDUSTRIEL
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION
DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES
(MATHÉMATIQUES APPLIQUÉES)
DÉCEMBRE 2016

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé :

RÉ-OPTIMISATION D'HORAIRES DE PERSONNEL EN AJOUTANT DES
TRANSFERTS ENTRE DÉPARTEMENTS

présenté par : SOUISSI Sarra

en vue de l'obtention du diplôme de : Maîtrise ès sciences appliquées

a été dûment accepté par le jury d'examen constitué de :

Mme LAHRICHI Nadia, Ph. D., présidente

M. DESAULNIERS Guy, Ph. D., membre et directeur de recherche

M. SOUMIS Francois, Ph. D., membre et codirecteur de recherche

M. EL HALLAOUI Issmaïl, Ph. D., membre

DÉDICACE

*À ma famille et tous ceux que j'aime
merci de m'avoir toujours soutenu.*

REMERCIEMENTS

Au terme de ce projet, j'aimerais exprimer mes sentiments de gratitude aux différentes personnes qui m'ont encadrées et soutenues tout au long de mon travail rendant ainsi possible l'accomplissement du présent projet.

Je tiens tout particulièrement à exprimer ma profonde gratitude et respect au professeur Guy Desaulniers, mon directeur de recherche, pour ses conseils précieux et son aide tant morale que financière tout au long de ce travail. Je souhaite remercier infiniment mon co-directeur de recherche le professeur François Soumis. Merci pour son soutien et son appui qui m'ont permis d'achever ce projet. C'est à la fois un privilège et une expérience exceptionnelle que d'avoir pu bénéficier de leurs conseils.

Je voudrais remercier Nadia Lahrichi d'avoir acceptée de présider mon jury de maîtrise, ainsi qu'Issmaïl El Hallaoui d'avoir accepté d'en faire partie.

Je tiens à remercier également et à témoigner toute ma reconnaissance à mes collègues au Gerad, pour l'expérience enrichissante et pleine d'intérêt qu'ils m'ont fait vivre durant ma maîtrise.

A tous ceux qui, de près ou de loin, directement ou indirectement, ont contribué à la concrétisation des objectifs visés par ce travail, je dis merci.

RÉSUMÉ

Le problème de planification d'horaires du personnel prend de plus en plus de place dans les problèmes traités en recherche opérationnelle. La littérature sur ce problème présente une vaste collection de modèles et de méthodologies de résolution. Ce problème consiste à déterminer un horaire qui spécifie le début et la fin de chaque quart de travail, les durées et les positions de pauses durant un quart de travail sur un horizon de planification. En général, l'horizon de planification est discrétisé en un ensemble de périodes de durées égales et une demande est définie en fonction de ces périodes. La demande représente le nombre d'employés requis pour chaque période.

Pour couvrir la demande, il faut affecter un nombre d'employés pour chacune des périodes de l'horizon de planification tout en respectant la réglementation de la convention collective. Il n'est pas toujours possible de faire coïncider l'horaire ainsi obtenu avec la demande. Il se peut que le nombre d'employés affectés pour une période donnée soit supérieur au nombre d'employés demandés, dans ce cas on dit qu'il y a sur-couverture de la demande. Parfois, le nombre d'employés affectés pour une période donnée est inférieur au nombre d'employés demandés, et dans ce cas, c'est une sous-couverture de la demande. Le but est d'éviter de créer de la sous-couverture pour pouvoir assurer une bonne qualité de service ou de créer de la sur-couverture pour éviter de payer des employés pour un temps de travail improductif. Plusieurs particularités peuvent être observées quand nous considérons des problèmes réels spécifiques. Par exemple, le fait que la structure d'entreprise se base sur un ensemble d'unités distinctes (des départements) et que pour chaque unité un certain nombre d'employés peut y être rattachés. De plus, de par leurs expériences, les employés peuvent travailler dans plusieurs départements.

En pratique, dans le contexte multi-départements, les horaires sont d'abord faits séparément pour chaque département. Ensuite, les employés qui sont en surplus dans un département peuvent éventuellement être affectés à d'autres départements pour combler les manques du personnel s'ils sont qualifiés. Il serait toutefois préférable de planifier les horaires de façon globale. Nous avons commencé par développer un modèle global pour avoir un horaire centralisé pour tous les départements en une seule étape. Mais, ce modèle devient difficile à résoudre quand la taille augmente, voire même impossible dans certains cas.

Ce mémoire présente une méthode heuristique de résolution du problème de planification d'horaires du personnel dans un contexte multi-départements avec la possibilité de partager la main-d'œuvre qualifiée entre les départements. Nous considérons qu'une entreprise a un

ensemble de départements et un ensemble d'employés et que chaque département se spécialise dans la production d'un bien ou l'offre d'un service. Un employé peut avoir plusieurs qualifications. Cependant, il est rattaché à un département appelé département d'origine. En plus, de par son expérience, un employé peut travailler occasionnellement dans d'autres départements secondaires. Chaque département est mono-activité, il a donc une seule courbe de demande qui représente le nombre d'employés requis pour chacune des périodes de l'horizon de planification. L'heuristique proposée se déroule en deux phases. La première phase est une résolution du problème mono-département pour chacun des départements sans permettre le partage de la main-d'oeuvre. La démarche classique est de générer l'ensemble global des quarts par département sans prise en compte des possibilités de transfert et de chercher à avoir la meilleure solution par département. Parfois les employés rattachés à chaque département ne suffisent pas pour répondre à la demande. La deuxième phase a une vue plus globale. C'est une ré-optimisation de la solution obtenue dans la première phase en ajoutant des transferts. Certaines règles de transfert sont à respecter comme par exemple la possibilité d'avoir au plus un transfert par jour par employé.

Dans ce projet, nous avons commencé par résoudre le problème d'une façon globale en générant tous les types de quarts possibles. La solution ainsi obtenue est la solution optimale globale du problème. Ensuite pour la deuxième phase de l'heuristique, trois approches de génération de quarts avec transfert sont proposées. Dans la première approche, nous interdisons la modification de l'horaire issu de la première phase. Nous nous permettons d'ajouter des quarts avec transfert sans remettre en question les quarts retenus dans la solution de la première phase. Cette approche est très restrictive, elle ne permet pas une grande amélioration au niveau de la solution obtenue dans la première phase. Puis, dans le but d'améliorer cette solution, la modification de la solution initiale de la première phase est autorisée en transformant les quarts retenus en des nouveaux quarts avec transferts d'une façon plus flexible. La transformation peut se faire en allongeant ou en raccourcissant un quart avec l'ajout des périodes de transfert. Nous obtenons de meilleurs résultats que ceux de la première approche mais la solution globale optimale n'est atteinte que pour une seule instance de test. Dans le but d'atteindre cette solution, une troisième approche est proposée. Pour cette approche, nous nous donnons encore plus de liberté pour modifier la solution de première phase. Nous pouvons permuter les quarts des employés entre eux pour favoriser les transferts. Nous obtenons de bonnes solutions avec des temps de calcul raisonnables. Nous considérons que ces solutions peuvent être améliorées avec l'amélioration de la solution de la première phase. Vers la fin du mémoire, une attention particulière est portée sur l'amélioration de la première phase classique. Une nouvelle méthode pour résoudre la première phase est proposée. Elle permet de donner une meilleure solution de première phase plus adaptée à l'ajout des

transferts. Les résultats obtenus sont nettement améliorés et nous avons réussi à obtenir des solutions avec moins de 5% d'écart avec la solution optimale globale dans 75% des instances de test en des temps de calcul raisonnables.

ABSTRACT

Problems involving staff schedules are becoming increasingly important in operational research. In recent decades, these problems have been widely studied in the literature and several methods have been described. They aim to build a work schedule for employees over a certain length of time that is called planning the horizon. A set of periods of equal duration cover the planning the horizon. A demand is defined according to these periods, representing the number of employees required for each period.

Feasible shifts are assigned to employees to satisfy this given demand while respecting the collective agreements. Sometimes, it is not possible to meet the demand exactly. The number of employees assigned for a given period may be greater than the number of employees requested. We can say that we have an over-coverage of the demand at this period. Sometimes the number of employees assigned for a given period is less than the number of employees requested. In this case, there is demand under-coverage. We want to avoid creating under-coverage in order to ensure a good quality of service and creating over-coverage for paying unproductive working time.

Different extensions of the classical personnel scheduling problem can be observed when specific real problems are considered. For example, if employees have multiple qualifications and can work on multiple jobs. In this case, it is important to know which employee is assigned to which job at a given time. In this project, we focus on building schedules for employees in large companies. Generally, large companies are divided into different departments and their workload peaks do not necessarily occur at the same time in all the departments. To avoid hiring new employees, these companies train employees to work in different departments. Each employee has a primary qualification and other qualifications. When an employee works outside the department of his primary qualification, we say that he/she is transferred. Certain rules such as limiting the number of transfers per day to one restrict the employee transfers.

In practice, in the multi-department context, schedules are first made separately for each department. Then, employees who are not working in their original department and are qualified to work in other departments can be assigned to other departments to fill the gaps of the demand. However, it would be preferable to plan schedules in a global way. We started by developing a global model that allows to build a schedule for all departments in one step. But, this model is difficult to solve when the size of the problem increases, or even impossible in some cases.

This paper presents a heuristic method for solving the problem of building staff schedules in a multi-department context with the possibility of sharing the workforce between the departments. More specifically, we are interested in transferring employees between departments. The proposed solution approach addresses the problem in two phases. Both phases are based on integer programming which is used to determine the assignment of each employee. The first phase solves a single-department problem for each department separately, without allowing the sharing of the workforce, i.e., without considering any transfer. Shifts are generated by department and we consider the employees that have this department as their original. This limits the number of shifts compared to the number of shifts for the global model. The second phase detects the under-coverage in each department and generates new potential shifts with transfers. It is a re-optimization of the solution obtained in the first phase with the addition of transfers.

In this project, we begin by solving the problem in a global way by generating all types of possible shifts. The solution thus obtained is the optimal overall solution of the problem. Then for the second phase of the heuristic, three approaches are developed. In the first one, we don't allow the modification of the schedule resulting from the first phase. We add shifts with transfers without modifying the shifts of the first phase solution. This approach is very restrictive and it does not allow a great improvement of the solution obtained in the first phase. Then, in order to improve this solution, we can modify the initial solution of phase one by transforming the retained shifts into new shifts with transfers in a more flexible way. Transformations can be done by extending or reducing the shift length and adding some transfer periods. We obtain better results than those of the first approach but the optimal overall solution is only reached for a single test case. In order to achieve this solution, a third approach is proposed. For this approach, we add more flexibility to the shift generation method and we allow the modification of the first-phase solution by swapping employee shifts. We get good solutions in reasonable computing times. We consider that these solutions can be improved if we have better solutions of the first phase. Thus, towards the end of the thesis, special attention is paid to the improvement of the first phase. A new method to solve the first phase is proposed. The results obtained are improved and we manage to obtain solutions with less than 5% gap from the overall optimal solution in 75% of the test cases in reasonable computing times.

TABLE DES MATIÈRES

DÉDICACE	iii
REMERCIEMENTS	iv
RÉSUMÉ	v
ABSTRACT	viii
TABLE DES MATIÈRES	x
LISTE DES TABLEAUX	xii
LISTE DES FIGURES	xiv
LISTE DES ANNEXES	xv
CHAPITRE 1 INTRODUCTION	1
1.1 Contexte du problème étudié	1
1.2 Éléments de la problématique	2
1.3 Objectifs de recherche	3
1.4 Plan du mémoire	4
CHAPITRE 2 REVUE DE LITTÉRATURE	5
2.1 Problème de planification d’horaires du personnel	5
2.1.1 Décomposition du problème de planification d’horaires du personnel	6
2.1.2 Classification selon la méthode de sélection des quarts	8
2.1.3 Classification selon la nature de la main d’œuvre	10
2.1.4 Classification selon le domaine d’application	11
2.2 Problème de planification d’horaire du personnel dans un contexte multi-départements	11
2.2.1 Contexte multi-départements	12
2.2.2 Heuristique en deux phases	12
CHAPITRE 3 DÉFINITION DU PROBLÈME ET MODÈLE GLOBAL	14
3.1 Énoncé du problème multi-départements	14
3.2 Génération de quarts	17

3.3	Modèle global	19
3.3.1	Notation	19
3.3.2	Formulation mathématique	21
3.4	Méthode de résolution du modèle global	22
CHAPITRE 4	MÉTHODE DE RÉOLUTION EN DEUX PHASES	23
4.1	Première phase : résolution département par département	23
4.1.1	Énoncé du problème mono-département	23
4.1.2	Génération des quarts internes	23
4.1.3	Formulation mathématique du modèle de la première phase	24
4.2	Deuxième phase : ré-optimisation en permettant les transferts	24
4.2.1	Énoncé du problème de ré-optimisation	25
4.2.2	Génération des quarts avec transfert	25
4.2.3	Formulation mathématique	35
CHAPITRE 5	ANALYSE DES RÉSULTATS	36
5.1	Environnement de test	36
5.1.1	Présentation des instances de test	36
5.1.2	Valeurs des paramètres et constantes du problème	38
5.2	Le nombre de quarts générés par instance	39
5.3	Les critères de performance	41
5.4	Résultats du modèle global	42
5.5	Résultats de l'heuristique en deux phases	43
5.5.1	Résultats de la première phase	43
5.5.2	Résultats de la phase de ré-optimisation	44
5.5.3	Heuristique en deux phases avec la première phase optimale	46
5.6	Amélioration de la première phase	47
5.6.1	Nouvelle approche de la première phase : Temps de calcul rapide	48
5.6.2	Nouvelle approche de la première phase : Temps de calcul allongé	49
CHAPITRE 6	CONCLUSION	53
6.1	Synthèse des travaux	53
6.2	Améliorations futures	54
RÉFÉRENCES	55
ANNEXES	57

LISTE DES TABLEAUX

Tableau 3.1	Notations relatives à un quart de travail	18
Tableau 4.1	Caractéristiques de l'instance IA(2,4)	28
Tableau 4.2	Les quarts internes retenus pour l'instance IA(2,4)	29
Tableau 4.3	Les quarts de transfert possibles pour la première approche du modèle de ré-optimisation	30
Tableau 4.4	Caractéristiques de l'instance IA(2,3)	33
Tableau 4.5	Les quarts internes retenus pour l'instance IA(2,3)	33
Tableau 5.1	Les instances de test de petite taille générées aléatoirement	38
Tableau 5.2	Les grandes instances de test générées aléatoirement	38
Tableau 5.3	Nombre de quarts générés pour le modèle global	40
Tableau 5.4	Comparaison du nombre de quarts total entre le modèle global, la pre- mière phase et les trois approches de la phase de ré-optimisation . . .	41
Tableau 5.5	Résultats numériques du modèle global pour toutes les instances . . .	43
Tableau 5.6	Résultats numériques de la première phase de l'heuristique en deux phases	44
Tableau 5.7	Résultats de la phase de ré-optimisation avec les différentes approches	45
Tableau 5.8	Comparaison des temps de calcul	46
Tableau 5.9	Résultats numériques du modèle global pour les instances de petite taille	48
Tableau 5.10	Nouvelle approche de la première phase : Temps de calcul rapide . . .	49
Tableau 5.11	Résultats de la phase de ré-optimisation pour la nouvelle approche de la première phase (Temps de calcul rapide)	50
Tableau 5.12	Nouvelle approche de la première phase : Temps de calcul allongé . .	50
Tableau 5.13	Résultats de la phase de ré-optimisation pour la nouvelle approche de la première phase (Temps de calcul allongé)	51
Tableau A.1	Nombre de quarts générés par instance pour le modèle de la première phase	57
Tableau A.2	Nombre de quarts générés par instance pour le modèle de ré-optimisation sans remise en question des quarts internes	57
Tableau A.3	Nombre de quarts générés par instance pour le modèle de ré-optimisation avec remise en question des quarts internes	58
Tableau A.4	Nombre de quarts générés par instance pour le modèle de ré-optimisation avec permutation d'employés	58
Tableau B.1	Résultats de la première phase de l'heuristique en deux phases	59

Tableau B.1	Résultats de la première phase de l’heuristique en deux phases (suite)	60
Tableau B.1	Résultats de la première phase de l’heuristique en deux phases (suite et fin)	61
Tableau B.2	Résultats de la première approche de la phase ré-optimisation de l’heuristique en deux phases	61
Tableau B.3	Résultats de la deuxième approche de la phase ré-optimisation de l’heuristique en deux phases	62
Tableau B.4	Résultats de la troisième approche de la phase ré-optimisation de l’heuristique en deux phases	62
Tableau C.1	Résultats de la deuxième phase adaptée à la nouvelle première phase (Temps de calcul Rapide)	63
Tableau C.2	Résultats de la deuxième phase adaptée à la nouvelle première phase (Temps de calcul allongé)	63

LISTE DES FIGURES

Figure 3.1	Exemple de courbe de demande pour le département d	15
Figure 3.2	Exemple d'horaire avec deux employés et deux départements	16
Figure 4.1	Courbes de demande des deux départements de l'instance IA(2,4) sur une journée	28
Figure 4.2	Solution de la première phase pour l'instance IA(2,4)	29
Figure 4.3	Solution pour l'instance IA(2,4) en utilisant la première approche de la phase de ré-optimisation	30
Figure 4.4	Solution pour l'instance IA(2,4) en utilisant la deuxième approche de la phase de ré-optimisation	31
Figure 4.5	Courbes de demande des deux départements de l'instance IA(2,3) pour une journée	33
Figure 4.6	Solution de la phase de ré-optimisation avec la deuxième approche pour l'instance IA(2,3)	34
Figure 4.7	Solution de la phase de ré-optimisation avec la troisième approche pour l'instance IA(2,3)	35
Figure 5.1	Comparaison des écarts des solutions de l'heuristique en deux phases classique et la nouvelle solution	52
Figure 5.2	Comparaison des temps de résolution des solutions de l'heuristique en deux phases classique et la nouvelle solution	52

LISTE DES ANNEXES

ANNEXE A	DÉTAILS SUR LE NOMBRE DE QUARTS GÉNÉRÉS	57
ANNEXE B	RÉSULTATS DE L'HEURISTIQUE EN DEUX PHASES	59
ANNEXE C	RÉSULTATS DE L'HEURISTIQUE EN DEUX PHASES AVEC LA NOUVELLE APPROCHE DE LA PREMIÈRE PHASE	63

CHAPITRE 1 INTRODUCTION

La réussite d'une entreprise dépend pour beaucoup de sa capacité à gérer ses employés. Pour pouvoir acquérir de nouvelles parts de marché et augmenter leurs chiffres d'affaire, les entreprises doivent innover et lancer de nouveaux produits ou services sur le marché. Ainsi le nombre d'employés peut vite augmenter de façon à devenir ingérable sans outils spécialisés. La construction manuelle des emplois du temps des ressources humaines dans les entreprises de grande taille est devenue difficile et parfois même impossible parce qu'il est important de déterminer de façon précise non seulement les jours de travail et de congé de chaque employé mais aussi les heures de début de travail, les durées de travail, les pauses repas et les pauses café durant le temps de travail de chaque employé. Pour cela, les organisations de grande taille doivent avoir des outils performants pour évaluer leurs besoins en main-d'œuvre et pour organiser le travail des employés en fonction des qualifications de chacun. De plus, la charge salariale des employés est une principale composante des dépenses de l'entreprise. De nos jours, de plus en plus d'intérêt est porté sur le problème de planification d'horaires du personnel. Le but est d'avoir un horaire qui satisfait la demande en nombre d'employés tout en respectant les règles de la convention collective et les contraintes liées au domaine d'application. Parmi ces domaines, nous pouvons citer par exemple, le transport terrestre ou aérien, les unités d'urgence (les policiers, les pompiers, les secours), les hôpitaux, les centres d'appel ou encore les magasins de vente de détail. Dans notre projet, nous nous intéressons au problème de planification d'horaires du personnel pour les magasins de vente de détail. Pour ces entreprises, les dépenses liées aux ressources humaines sont les plus coûteuses. Et puisque ces enseignes sont en concurrence continue pour la baisse des prix, la diminution des charges aura un impact direct sur les prix des produits offerts aux consommateurs et des parts de marché des entreprises.

1.1 Contexte du problème étudié

Nous considérons qu'une des principales composantes qui permet de réduire considérablement les coûts est la composante des coûts liés à la main-d'œuvre. Il faut donc affecter les bons employés au bon moment et éviter d'affecter des employés qui resteront inactifs quand la charge de travail baisse ou avoir un manque d'employés quand la charge de travail augmente. Il faut trouver le bon équilibre entre la charge de travail à effectuer et les employés nécessaires pour la satisfaire, tout en essayant de maximiser la satisfaction des employés.

Le domaine de la vente de détail a la particularité de traiter un volume important de marchan-

disent en fournissant des services connexes pour pouvoir offrir aux clients une vaste gamme de produits et ainsi faire des économies d'échelle. Le but est d'attirer une grande clientèle diversifiée. Pour pouvoir satisfaire cette clientèle, il faut avoir des employés qualifiés pour répondre à leurs besoins. Ces magasins de vente de détail à rayons multiples peuvent être divisés en plusieurs départements. Chaque département est responsable d'un ensemble de produits comme par exemple le département des produits alimentaires ou le département des produits électroniques ou des produits de maison. Le besoin en nombre d'employés varie dans le temps pour chaque département et le nombre d'employés est assez élevé d'où le besoin d'avoir des outils efficaces pour résoudre le problème de planification d'horaires du personnel dans le cadre d'une entreprise avec plusieurs départements et la possibilité de partager la main-d'œuvre entre les départements au besoin. L'objectif des problèmes de planification d'horaires du personnel est de sélectionner un ensemble de quarts de travail pour chaque employé tout en respectant un certain nombre de règles.

Dans notre projet, nous considérons un ensemble de départements et un ensemble d'employés. Chaque employé est rattaché à un département. D'après son expertise et son ancienneté, il peut travailler dans d'autres départements. L'entreprise voudrait construire un horaire de façon centralisée qui prend en compte les qualifications de chaque employé et qui répond aux exigences en termes de ressources humaines pour tous les départements simultanément tout en respectant les clauses de la convention collective de travail. Des règles concernant le partage de la main-d'œuvre peuvent être exprimées par l'entreprise comme par exemple l'importance d'avoir un nombre maximal de déplacements par jour pour chaque employé entre les départements où il peut travailler.

Ce projet a été lancé par une société de développement de logiciel qui aide les entreprises à contrôler le coût de leur main-d'œuvre et à améliorer la productivité de leurs employés dans différents secteurs d'activités. Il est important d'avoir une solution qui permet de résoudre le problème de planification d'horaires du personnel dans des temps raisonnables qui pourra être intégrée au logiciel commercialisé par cette entreprise.

1.2 Éléments de la problématique

Afin de mieux comprendre notre problématique, il est important de définir les informations nécessaires pour pouvoir appliquer notre méthode et les résultats que nous désirons obtenir. Dans notre cas, nous sommes dans un environnement à plusieurs départements où le travail est continu c'est-à-dire qu'il n'y a pas d'interruption, par exemple le soir où la fin de semaine. Nous voulons établir un horaire pour tous les employés sur une durée d'une semaine. De plus, chaque employé doit avoir deux jours de congé successifs. Nous considérons que la demande

pour chaque département est déjà fournie et que les jours de congés des employés sont pré-assignés dans un niveau décisionnel antérieur. Des informations concernant chaque employé sont fournies comme les départements où il peut travailler, son taux horaire de travail et ses disponibilités sur la période de planification. Des règles spécifiques sont appliquées sur les quarts de travail relatives à leurs débuts et à leurs durées. En effet, pour assurer une bonne qualité de vie aux employés, il faut éviter d’avoir des quarts trop longs qui fatiguent et peuvent engendrer des absences et des maladies mentales ou physiques. Les quarts trop courts sont aussi à éviter. Il faut donc avoir une durée minimale et une durée maximale pour chaque quart de travail. Si nous discrétisons le temps global de planification en un ensemble de périodes égales et que nous estimons la demande en fonction de ces périodes, il faut affecter un certain nombre d’employés à chacune des périodes pour couvrir la demande. Si le nombre d’employés affectés est supérieur au nombre d’employés demandés, une sur-couverture est créée. Si le nombre d’employés affectés est inférieur au nombre d’employés demandés, une sous-couverture est créée. Le but est d’éviter de créer de la sous-couverture pour pouvoir assurer la qualité de service requis ou de créer de la sur-couverture pour ne pas payer des employés pour un temps de travail improductif.

Les responsables de ressources humaines ont l’habitude de construire l’horaire de travail pour chaque département sans prendre en compte les demandes des autres départements. Ils ont tendance à embaucher de nouvelles ressources propres à chaque département ou de faire travailler leurs ressources en temps supplémentaire. Nous sommes dans un contexte multi-départements et nous voulons avoir un horaire généralisé pour tous les départements. Dans ce contexte où le partage de la main-d’œuvre est admissible, il est plus rentable de solliciter des ressources disponibles et capables de faire le travail que d’embaucher de nouvelles ressources ou de payer des heures supplémentaires pour répondre à la demande de chaque département.

1.3 Objectifs de recherche

Nous proposons une méthode de résolution du problème de planification d’horaires du personnel pour les entreprises de grande taille en permettant de transférer occasionnellement des employés d’un département à l’autre. Dans ce mémoire, nous proposons une décomposition du problème en deux phases pour obtenir des temps de calcul raisonnables. Nous commençons par résoudre le problème de planification d’horaires du personnel pour chaque département sans tenir compte du fait qu’il est possible de partager la main-d’œuvre. Nous obtenons un horaire optimisé pour chaque département auquel nous ajoutons les transferts en deuxième phase. Dans ce projet, c’est l’amélioration de la deuxième phase qui nous intéresse. Il faut respecter les règles de la convention collective de travail et les règles de déplacement des

employés entre les départements. Notre objectif est de développer un modèle mathématique pour chaque étape du processus que nous résolvons avec un optimiseur commercial comme FICO Xpress-MP. Pour valider nos modèles, nous vérifierons l'efficacité de notre méthode sur des instances de différentes tailles. Il est important d'établir des critères de validation de la qualité de la solution comme la valeur de la fonction objectif où le nombre de sous-couvertures et de sur-couvertures. De plus, il faudrait montrer que la méthode est efficace et qu'il est possible d'obtenir une bonne solution dans des temps de calcul acceptables.

1.4 Plan du mémoire

La suite de ce document se présente comme suit. Le chapitre 2 est une revue de littérature qui aborde les problèmes de planification d'horaires du personnel d'abord dans un cadre général ensuite dans le contexte multi-départements. Ce problème a largement été étudié dans la littérature et plusieurs revues ont été publiées sur le sujet. Le cas multi-départements et spécifiquement la décomposition en deux phases seront étudiés à la fin de ce chapitre.

Dans le chapitre 3, nous introduisons le modèle global de Munezero (2014) que nous utilisons pour le problème de planification d'horaires du personnel dans un contexte multi-départements lorsque chacun des départements est supposé être mono-activité. Le modèle mathématique ainsi que la méthode de génération de quarts et la méthode de résolution sont abordés dans ce chapitre. Par la suite, nous montrons que parfois il n'est pas possible de résoudre ce modèle sur des instances de grande taille, d'où le recours à une méthode de décomposition en deux phases. La méthode en deux phases est proposée et expliquée dans le chapitre 4. Dans ce chapitre, nous commençons par introduire brièvement la première phase qui est une phase de résolution "département par département" sans considérer les transferts. La deuxième partie du chapitre est consacrée à la résolution de la deuxième phase qui est une ré-optimisation du problème en se basant sur la solution de la première phase. Nous exploitons différentes façons de générer les quarts de transfert pour résoudre la phase de ré-optimisation. Nous présentons les différents jeux de données auxquels nous avons eu accès dans le chapitre 5. Nous analysons ensuite les résultats pour les approches de résolution de la deuxième phase en comparaison aux résultats du modèle global. Au dernier chapitre, nous résumons les travaux réalisés tout au long du projet en précisant les améliorations à effectuer dans le futur.

CHAPITRE 2 REVUE DE LITTÉRATURE

Les problèmes de planification d’horaires du personnel ont été largement étudiés et ceci est probablement dû au besoin des entreprises d’optimiser leurs coûts, le coût lié à la main d’œuvre étant considéré comme une composante importante des coûts directs. Dans notre projet, nous étudions le problème de planification d’horaires du personnel dans un contexte multi-départements. Notre but est d’avoir une méthode pour résoudre ce problème sur un ensemble de départements de différentes tailles dans des temps raisonnables. Chaque département est responsable d’une seule activité et il est possible de partager les employés entre les départements. Dans ce chapitre, nous commençons d’abord par étudier les travaux déjà réalisés sur le problème de planification d’horaires du personnel dans un cadre général et les différentes classifications de ce problème. Ensuite, nous explorons les études qui se rapprochent le plus de notre problématique, notamment le cas multi-départements et la décomposition en deux phases.

2.1 Problème de planification d’horaires du personnel

Les approches proposées dans la littérature pour le problème de planification d’horaires du personnel aujourd’hui sont différentes de celle introduite par Dantzig (1954), une des premières applications de la programmation mathématiques dans le domaine de la planification d’horaires. Le modèle de Dantzig (1954) résout le problème proposé par Edie (1954) avec un programme en nombre entiers. C’est une approche de recouvrement d’ensemble pour la planification d’horaires à des postes de péage afin de réduire le retard des effectifs et ainsi avoir un trafic plus fluide.

En général la planification d’horaires du personnel permet de déterminer le nombre d’employés nécessaire à chaque période de temps dans l’horizon de planification en spécifiant les périodes de pause pour chaque quart de travail et de repos pour chaque employé. Cela implique la détermination de la demande en nombre d’employés par période ainsi que les quarts de travail admissibles qui respectent différentes contraintes liées au type de problème. Le choix des quarts personnalisés de travail pour chaque employé peut devenir complexe si le problème admet un niveau de flexibilité élevé. Par exemple, nous pouvons prendre en compte les préférences horaires des employés, les types de quarts, les différentes activités qui peuvent être réalisées par chaque employé ainsi que les contraintes spécifiques à chaque industrie. D’après Ernst et al. (2004a), la solution pour surmonter la formulation à grande échelle est d’utiliser des techniques de décomposition et des algorithmes heuristiques. Dans la pro-

chaîne section, nous introduisons différentes décompositions du problème pour une meilleure résolution dans le cas où le problème devient difficile à résoudre avec un seul processus.

2.1.1 Décomposition du problème de planification d’horaires du personnel

Pour mieux résoudre le problème de planification d’horaires, il faut à la fois utiliser des outils d’aide à la décision avec des algorithmes performants et d’autre part, diviser le problème en sous-problèmes plus faciles à résoudre. Ceci permet au processus de résolution d’être plus efficace. Dans cette partie, nous abordons la décomposition du problème en sous-problèmes qui peuvent être dépendants ou indépendants les uns des autres.

Une des premières méthodes de classification a été proposée par Baker (1976). Selon lui, trois catégories peuvent être retenues. La première catégorie est le problème de construction des quarts anonymes dit “*Shift scheduling*” où une planification jour par jour est construite. La deuxième catégorie est le problème de planification des journées de congé ou “*days off scheduling problem*”. Ce type de problème prend place dans un environnement de travail continu et cyclique. Pour Baker (1976), il s’agit d’une planification sur sept jours par semaine où chaque employé peut travailler jusqu’à cinq jours. Il faut donc attribuer au moins deux jours de congé pour chaque employé. Et enfin, la troisième catégorie comporte le jumelage des deux premières catégories. C’est le problème de construction des cycles de travail “*tour scheduling problem*”. Ce problème cherche à obtenir simultanément les jours de travail et les jours de repos ainsi que les quarts spécifiques par jour pour chaque employé. La taille du problème peut vite augmenter avec le nombre de règles de travail et la nature de la main d’œuvre. La résolution devient alors très complexe.

Ensuite, Ernst et al. (2004a) proposent une revue sur les applications, méthodes et modèles des problèmes de planification d’horaires du personnel. Il admet que le processus global de résolution du problème peut être réparti en différents sous-problèmes. D’après lui, la résolution de ces sous-problèmes engage six modules qui selon le problème à résoudre peuvent être regroupés ou pas. En liaison avec le domaine d’application spécifique, un ou plusieurs modules peuvent être pris en considération.

1. Modélisation de la demande “*Demand modeling*” : Comme il est souvent difficile de connaître la demande exacte en nombre d’employés par période de l’horizon de planification, il faut donc commencer par faire une étude pour estimer l’effectif nécessaire afin d’avoir une bonne qualité de service. Ceci peut se traduire par l’utilisation des données historiques ou par des méthodes de simulation et de prévision de la demande. La demande peut être dite continue quand le travail se fait sur 24h par jour ou dis-

continue quand il y a des heures où il n’y a pas de travail.

2. Planification des journées de congé “*Days off scheduling*” : Généralement la durée de l’horizon de planification ne coïncide pas avec le nombre de jours de travail des employés. En effet, sur un horizon de travail de sept jours, les employés travaillent cinq jours la plupart du temps. Il est donc nécessaire de spécifier pour chaque employé deux jours de repos.
3. Génération et sélection des quarts anonymes “*Shift scheduling*” : Ce module a pour objectif de générer un grand ensemble de quarts anonymes admissibles. Pour qu’un quart soit admissible, il faut qu’il prenne en compte les règles de la convention collective comme les débuts admissibles ou les durées admissibles s’il y a lieu. A cette étape du processus, nous ne spécifions pas les particularités liées à chaque employé. L’objectif de ce module est de sélectionner le sous-ensemble optimal de quarts qui permet de répondre à la demande et le nombre d’employés requis pour chaque quart.
4. Construction de lignes de travail “*Line of work construction*” : Parfois, nous pouvons regrouper un sous-ensemble de quarts pour couvrir la durée de l’horizon de planification. Pour cela, il faut prendre en compte les règles de la convention collective comme par exemple le temps minimal de repos requis entre deux quarts consécutifs. Dans certains domaines d’application, ce module peut ne pas être utilisé et les quarts retenus dans le module précédent servent de données d’entrée pour le module suivant.
5. Affectation des activités aux quarts “*Task assignment*” : Dans certains contextes, une ou plusieurs activités peuvent être affectées à un quart ou une ligne de travail. Cela dépend de la demande et des qualifications des employés.
6. Affectation des employés aux quarts “*Staff assignment*” : Finalement, il faut affecter des employés spécifiques pour chacun des quarts anonymes ou des lignes de travail retenus. Il faut donc prendre en compte les jours de congé de chaque employé ainsi que les qualifications pour les différentes activités. Ce module peut être associé au module de sélection des quarts anonymes ou de construction de lignes de travail. Cela permet de sélectionner des quarts personnalisés ou de construire des lignes de travail personnalisées.

Dans notre projet, nous nous intéressons à la génération de quarts et à la sélection des quarts personnalisés. On suppose que les jours de congé sont connus d’avance. Dans la prochaine section, nous nous intéressons aux différentes méthodes de résolution proposées dans la littérature.

2.1.2 Classification selon la méthode de sélection des quarts

La littérature sur la planification d'horaires du personnel présente une vaste collection de modèles et de méthodologies de résolution. La résolution du problème de planification d'horaires du personnel consiste à avoir un horaire qui spécifie le début et la fin de chaque quart de travail, les durées et les positions de pauses durant un quart de travail. D'après Van den Bergh et al. (2013), un grand nombre de travaux traitent de la programmation mathématique que ce soit la programmation en nombre entiers, la programmation linéaire, la programmation dynamique ou la programmation par objectif. Certains travaux utilisent des méthodes heuristiques et d'autres la programmation par contraintes. Il y a aussi des auteurs qui s'intéressent à des méthodes basées sur la simulation.

Parmi les méthodes de programmation mathématique, il existe des problèmes modélisés comme un programme linéaire, en nombre entiers ou mixte. La formulation de recouvrement d'ensemble qui a été introduite par Dantzig (1954) où les quarts sont générés explicitement reste très populaire dans le milieu scientifique. Cet ensemble est noté Q , l'ensemble des périodes de l'horizon de planification est noté I . Le paramètre $a_{i,q}$ prend la valeur de 1 si le quart $q \in Q$ contient la période $i \in I$ sinon 0. Pour chacune des périodes de l'horizon de planification, un nombre d'employés, noté b_i , est requis. c_q est le coût du quart q . La variable x_q prendra la valeur de 1 si le quart q est retenu dans la solution optimale et 0 sinon.

Le modèle se formule comme suit :

$$\min \sum_{q \in Q} c_q x_q \quad (2.1)$$

$$\sum_{q \in Q_e^i} a_{i,q} x_q \geq b_i \quad \forall i \in I \quad (2.2)$$

$$x_q \geq 0 \quad \forall q \in Q \quad (2.3)$$

L'objectif 2.1 du modèle est de sélectionner un sous-ensemble de quart parmi tous les quarts générés explicitement en minimisant le coût de travail total. Les contraintes 2.2 assurent que la demande est satisfaite à chaque période de l'horizon de planification et les contraintes 2.3 assurent l'intégralité et la non négativité des variables.

Cette formulation peut s'adapter à des besoins particuliers et cela en ajoutant les contraintes nécessaires. Parfois pour mieux s'adapter au besoin, les chercheurs ont tendance à ajouter un grand nombre de variables et de contraintes, ce qui rend le problème plus difficile à résoudre.

Jacobs et Bechtold (1993) ont démontré qu'il est possible de réduire le nombre d'employés

en ajoutant plus de flexibilité pour les début, les durées de quarts possibles et les possibilités de pause durant les quarts de travail. Plus de flexibilité permet de réduire le coût total de travail mais augmente la complexité du problème traité. En effet, Lau (1996) considère que le problème de planification d'horaires du personnel est un problème NP-difficile et ceci a été démontré pour une version restreinte de ce problème. C'est la raison qui explique les différentes heuristiques proposées dans la littérature qui permettent d'obtenir des solutions satisfaisantes pour des grands problèmes difficiles à résoudre d'une manière exacte.

D'après Brucker et al. (2011), la formulation mathématique en nombres entiers pour des problèmes de planification d'horaires du personnel peut être résolue avec un optimiseur de programmation linéaire comme par exemple, FICO Xpress-MP ou IBM-CPLEX.

Il a été montré qu'il est possible d'ajouter de la flexibilité à l'approche explicite qui se base sur le modèle de recouvrement d'ensemble de Dantzig (1954) en faisant varier les débuts de quarts, les durées et les périodes de pause pour chaque quart. En considérant cette approche explicite, il faudrait générer une variable pour chaque quart de travail avec toutes les possibilités de flexibilité. Il en résultera un large problème de recouvrement d'ensemble difficile à résoudre. Bechtold et Jacobs (1990) proposent un modèle implicite où les quarts sont classés en groupe selon leurs débuts, leurs durées et une fenêtre de temps durant laquelle une pause pourra être considérée. Les auteurs n'ont pas énuméré explicitement toutes les possibilités des pauses pour un quart donné mais ont introduit des variables supplémentaires. Ces variables permettent de choisir une seule pause pour chaque quart de travail dans la fenêtre de temps qui lui est associée. Bechtold et Jacobs (1996) ont démontré que cette modélisation implicite est équivalente à celle du modèle (2.1)-(2.3).

Le modèle implicite de Bechtold et Jacobs (1990) présume qu'il n'y a pas de chevauchement dans les fenêtres de temps des pauses pour les quarts avec le même début et la même durée. Addou and Soumis (2007) proposent de formuler le modèle implicite de Bechtold et Jacobs (1990) sans cette dernière hypothèse. Ils ajoutent un ensemble minimal de contraintes supplémentaires et les résultats montrent que cela n'entraîne pas une grande augmentation au niveau des temps de calcul.

Aykin (1996) a proposé une autre méthode implicite qui permet de modéliser la flexibilité des pauses de travail avec la possibilité d'avoir plusieurs pauses. Chaque pause peut débuter dans une fenêtre de temps différente. De plus, il a été montré que cette modélisation requière un plus petit nombre de variables que celle de Dantzig (1954). Un algorithme de branch-and-cut a été développé par Aykin (1998) pour résoudre la modélisation implicite proposée par Aykin (1996). L'algorithme ajoute des coupes et des limites supérieures pour les variables. Les résultats ont montré qu'il a été possible de résoudre le problème de planification d'horaires

du personnel pour des instances de grande taille à l’optimalité avec la modélisation de Aykin (1996) et l’algorithme de Aykin (1998).

Rekik et al. (2008) ont introduit deux types de pauses : les pauses standards qui sont indivisibles et les pauses fractionnaires qui peuvent être divisées. Les pauses standards ont une longueur fixée et peuvent commencer dans une fenêtre de temps déterminée. Contrairement aux pauses standards, les pauses fractionnaires peuvent être divisées en un ensemble de périodes de pauses sous certaines contraintes. Les auteurs ont montré que l’ajout de cette flexibilité a permis de réduire le nombre total d’employés.

Plus tard, Rekik et al. (2010) ont présenté deux approches de modélisation implicite pour le problème de planification d’horaires du personnel dans un environnement multi-jours continu. Les approches intègrent les pauses fractionnaires, les pauses standards, des fenêtres de temps et des restrictions sur les durées de travail avant et après les pauses. La première approche est inspirée du travail de Aykin (1996). La seconde est une extension de la formulation proposée par Bechtold et Jacobs (1990). Les résultats montrent que la première approche donne de meilleures solutions pour les instances de petite taille. Par contre, la deuxième trouve de bonnes solutions sur des instances difficiles à résoudre avec d’autres modèles.

D’autres auteurs se sont intéressés au problème de planification d’horaires du personnel. Nous pouvons citer les travaux de Côté et al. (2007), Côté et al. (2011a) et Côté et al. (2011b) où une utilisation des langages formels et de grammaires est introduite.

2.1.3 Classification selon la nature de la main d’œuvre

Souvent les activités à réaliser sont différentes les unes des autres et les employés peuvent être qualifiés pour certaines ou pour la totalité des activités traitées. Dans le premier cas, la main d’œuvre est dite hétérogène. D’après Alfares (2004), cela peut se traduire par le fait que des employés n’ont pas tous les mêmes qualifications, les mêmes jours de repos où les mêmes préférences horaires. Certains peuvent vouloir faire uniquement les quarts de jour par exemple, tandis que certains s’intéressent à faire les quarts de soir. De plus, dans la vie réelle, les employés possèdent des qualifications différentes, des niveaux d’expertise et des salaires différents. Dans le deuxième cas, la main d’œuvre est dite homogène. Cela prétend qu’il y a un seul type d’employés et que tous les employés ont les qualifications nécessaires pour effectuer l’ensemble des tâches requises.

2.1.4 Classification selon le domaine d'application

La gestion des effectifs est présente dans différents domaines d'application. Que ce soit dans le domaine du transport terrestre ou aérien, de la santé, de la sécurité publique ou de la vente au détail, il faut assurer une bonne qualité de service en minimisant les coûts liés au personnel. Pour chaque domaine, nous pouvons retrouver des modèles et des méthodes de résolution spécifiques. Dans cette section, nous expliquerons brièvement la spécificité liée à chaque domaine d'application.

1. Système de transport : c'est l'un des domaines les plus étudiés dans la littérature et le premier qui a été présenté dans les travaux de Dantzig (1954) et Edie (1954). Ce domaine regroupe la planification dans le transport aérien, transport des trains ou des lignes d'autobus. Ces types de problème ont la particularité qu'il faut prendre en compte des contraintes à la fois temporelles et spatiales comme le déplacement des équipages d'un aéroport à l'autre ou le déplacement des chauffeurs de train ou d'autobus. D'après Ernst et al. (2004b), les grandes compagnies aériennes ont même des départements dédiés à la gestion de leurs équipages.
2. Système de santé : Le problème le plus étudié dans le domaine de la santé est le problème d'horaires des infirmiers d'après Ernst et al. (2004a). Il faut avoir un nombre suffisant d'infirmiers qualifiés pour prendre soin des patients en tout temps.
3. Système de sécurité publique : Il comporte la planification d'horaires dans le service de la police, les ambulances et les pompiers. Cette catégorie est caractérisée par le fait que ce sont souvent des situations d'urgence qui y sont traitées. Il faut donc être capable d'avoir des systèmes très rapides qui distribuent les tâches en quelques secondes quand il y a différentes situations qui surviennent au même temps.
4. Système de vente au détail : Dans ce domaine, il est souvent question, actuellement, de gérer le personnel pour un magasin donné d'une façon indépendante des autres magasins. Toutefois cela commence à changer pour une gestion plus collaborative, c'est-à-dire, le partage des employés entre les différents sites.

Dans la section 2.2, nous allons introduire un aperçu des travaux réalisés dans le contexte multi-départements.

2.2 Problème de planification d'horaire du personnel dans un contexte multi-départements

Dans un contexte mono-département, résoudre le problème de planification d'horaires du personnel revient à construire un horaire de travail pour chaque employé en minimisant les

coûts et en essayant de couvrir la demande. Cela se fait sous les contraintes imposées par la convention collective. Dans le cas multi-départements, l'horaire doit couvrir la courbe de demande de chacun des départements tout en respectant la convention collective ainsi que des règles liées à la possibilité de partage des employés entre les départements. Dans la section 2.2.1, nous introduisons le contexte multi-départements et dans la section 2.2.2 c'est l'heuristique en deux phases qui sera abordée.

2.2.1 Contexte multi-départements

Dans les entreprises de grande taille, il est souvent question de traiter le problème de planification d'horaires des employés dans un environnement à plusieurs départements. La taille des départements peut varier d'un problème à l'autre. L'objectif est de répondre à la demande globale en spécifiant pour chaque période le nombre d'employés affectés à chaque département. La résolution de tous les départements dans un seul processus pourrait s'avérer difficile vu la taille du problème et du nombre de quarts générés. C'est ce qui explique le fait que dans la littérature notamment dans la proposition de thèse de Dahmen (2013) et dans le mémoire de Munezero (2014), la résolution se fait en deux étapes résolues d'une façon séquentielle.

Le contexte multi-départements prend son sens quand au moins un déplacement d'employé est proposé entre deux départements. Le déplacement se fait pour un employé qui n'est pas actif dans son département d'origine pour couvrir la sous-couverture dans un autre département où il y a un manque d'employés. Certainement, ce déplacement a un coût et des contraintes que nous appellerons "règles de transferts". Bard et Wan (2008) ont travaillé sur les règles qui doivent être respectées en cas de déplacement d'employé entre les départements. Un employé peut être transféré à un département si et seulement s'il est qualifié pour travailler dans ce département et qu'il n'est pas en service dans son département d'origine qu'il contribue à une sur-couverture dans son département d'origine.

D'après Dahmen (2013), la possibilité de déplacer les employés entre les départements ajoute un nouveau niveau décisionnel dans le processus. En effet, en plus de décider quand un employé travaille ou pas, il faut savoir dans quel département se déroule le travail et ceci pour chaque période travaillée.

2.2.2 Heuristique en deux phases

Dahmen (2013) et Munezero (2014) proposent des modèles assez semblables et qui se rapprochent le plus de notre problématique. Dans les deux ouvrages, nous retrouvons la notion de département d'origine ou département primaire et département de transfert ou départe-

ment secondaire. Chaque employé est affecté à un département primaire qui est défini par sa qualification primaire ou sa tâche principale et il peut être affecté à d'autres départements dépendamment de ses qualifications. Quand un employé travaille dans son département secondaire, on dit qu'il est transféré.

Dahmen (2013) propose une approche de résolution du problème de planification de quarts multi-jours multi-départements en deux étapes. La première étape est une étape d'agrégation temporelle où les données initiales sont agrégées et exprimées en fonction de périodes de plus grande durée. Par exemple, dans le cas où nous considérons des données temporelles en fonction d'une période de 15 minutes, c'est-à-dire que l'horizon de planification est discrétisé avec des périodes égales de 15 minutes, les données agrégées peuvent être exprimées en fonction des périodes de durée de 2h. La deuxième étape est une étape de désagrégation. En prenant en compte les résultats obtenus dans la première étape et en utilisant les données initiales désagrégées, il faut prendre les bonnes décisions de transferts pour couvrir les courbes de demandes des différents départements.

A l'opposé, dans le travail de Munezero (2014), il n'y a pas d'agrégation de la demande. Dans la première phase, le problème mono-département est traité pour chaque département à part. Cette étape permet d'avoir les quarts qu'un employé doit effectuer dans son département d'origine, c'est-à-dire sans prendre en compte les transferts entre les départements. Pour la deuxième phase, en se basant sur la solution de la première phase, des quarts avec transferts sont proposés. Deux approches ont été développées pour résoudre la deuxième phase, une résolution avec un algorithme glouton et une résolution avec un programme en nombres entiers. La programmation en nombres entiers s'est avérée la plus efficace des deux mais elle reste à améliorer. Dans son mémoire Munezero (2014), l'auteure propose une phase de ré-optimisation où elle ne remet pas en question les quarts obtenus de la solution de la première phase. Une remise en question des quarts de la première phase est prévue dans ce travail pour améliorer la qualité de la solution.

Concernant l'amélioration de la première phase, un travail est en cours de développement. En effet, Attia (2016) propose de modifier la première phase de Munezero (2014) pour qu'elle soit plus adaptée à l'ajout des transferts en deuxième phase. L'auteure s'intéresse au problème de planification d'horaires du personnel avec transferts inter-départementaux sur un horizon de planification continu d'une semaine. Chaque employé est rattaché à un département d'origine et peut être qualifié pour travailler dans d'autres départements secondaires. Les résultats préliminaires obtenus seront présentés vers la fin de ce mémoire.

CHAPITRE 3 DÉFINITION DU PROBLÈME ET MODÈLE GLOBAL

Comme nous l'avons mentionné dans la section 2.2, le problème étudié a été abordé dans le projet de maîtrise de Munezero (2014) avec une définition semblable à la nôtre. Dans le chapitre présent, nous commençons, dans la section 3.1, par introduire le problème de planification d'horaires du personnel dans un contexte multi-départements lorsque chacun des départements est mono-activité. Puis dans la section 3.2, nous détaillons la manière de générer les quarts pour notre modèle mathématique. Ensuite, nous présentons, à la section 3.3, le modèle mathématique qui permet de choisir un sous-ensemble de quarts parmi les quarts générés. Et finalement, nous décrivons la méthode de résolution du problème global dans la section 3.4.

3.1 Énoncé du problème multi-départements

Nous allons énoncer le problème ainsi que les notions sur lesquelles nous nous basons pour l'élaboration du modèle de la section 3.3. Les mêmes notions seront utilisées tout au long de ce mémoire.

Soit une entreprise avec un ensemble de départements D et un ensemble d'employés E . Chaque employé $e \in E$ est rattaché à un département d'origine d_e et peut être qualifié pour travailler dans un ensemble d'autres départements secondaires noté D_e . Si l'employé e est affecté à un département $d \in D_e$, on dit qu'il est transféré de d_e vers $d \in D_e$. Pour chaque département $d \in D$, nous pouvons définir l'ensemble d'employés internes E_d^{int} qui sont les employés dont le département d'origine est d et l'ensemble d'employés externes E_d^{ext} qui peuvent occasionnellement être transférés au département d . Chaque employé travaille principalement dans le département d'origine pour pouvoir effectuer son travail principal. Il peut être appelé à travailler dans un de ses départements secondaires pour accomplir un travail qu'il est capable de faire quand les employés internes du département secondaire sont insuffisants pour accomplir le travail requis. L'intervalle de temps durant lequel nous nous proposons de construire l'horaire s'appelle horizon de planification. Dans notre cas, l'horizon de planification est continu et discrétisé en un nombre de périodes de temps de durées égales. Le travail continu peut être observé dans les services d'urgence ou dans les entreprises qui offrent un service de 24h par jour. Soit I l'ensemble de toutes les périodes de l'horizon de planification et $i \in I$ une période de cet horizon. Les périodes $i \in I$ sont triées en ordre croissant où chacune est définie par son heure de début. Elles sont identifiées à l'aide d'entiers positifs de 1 à $|I|$. Si nous considérons que l'horizon est d'une semaine avec des

périodes de temps égales de 15 minutes chacune, nous pouvons définir l'ensemble ordonné $I = \{1, \dots, 672\}$ comme l'ensemble des 672 périodes qui commencent le jour 1 à la période $i = 1$ correspondant à l'intervalle de temps $[00h00; 00h15[$ et se terminent le jour 7 à la période $i = 672$. Chaque période couvre un intervalle de 15min, par exemple, la période i_{97} couvre l'intervalle de temps $[00h00; 00h15[$ du jour $j = 2$.

Soit J l'ensemble des jours qui constituent l'horizon de planification. Pour chaque jour, nous définissons l'ensemble des périodes qu'il contient comme $i \in I_j$, tel que ces périodes commencent le jour j à 00h00 ($i = 4 * 24 * (j - 1) + 1$) et se terminent à 23h45 ($i = 4 * 24 * j$).

Dans notre projet, nous considérons que l'horizon de planification est d'une semaine de sept jours ($|J| = 7$) mais que chaque employé doit avoir au moins deux jours de congé dont deux sont consécutifs. Les jours de congé sont pré-déterminés à un niveau décisionnel antérieur. Définissons J_e comme l'ensemble de jour de J où l'employé e peut travailler.

Chaque département $d \in D$ présente une courbe de demande indépendante de celles des autres départements. La courbe de demande est représentée par le nombre d'employés requis pour chaque période $i \in I$ de l'horizon de planification noté r_d^i . La courbe de demande peut être déduite des données historiques de l'entreprise ou bien construite par des méthodes de prévisions. Dans notre projet, nous considérerons que cette courbe est une donnée d'entrée au processus et qu'elle est déterministe. Habituellement, ce type de données d'entrée ne change pas en cours de résolution.

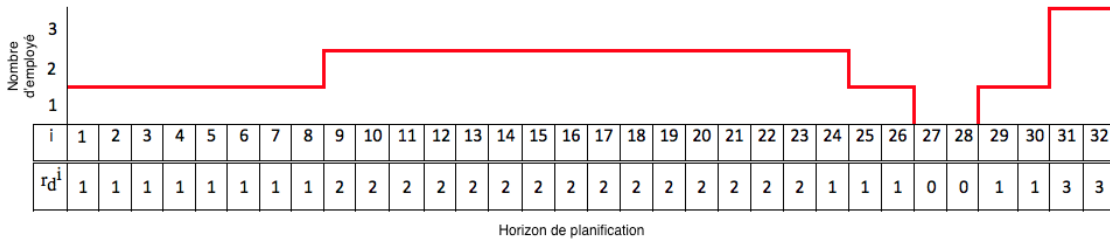


Figure 3.1 Exemple de courbe de demande pour le département d

La figure 3.1 présente un exemple de courbe de demande de la période $i = 1$ à la période $i = 32$, ce qui correspond aux 8 premières heures de l'horizon de planification, c'est-à-dire l'intervalle de temps $[00h00; 08h00[$ au jour 1.

Les quarts que nous utilisons sont des quarts mono-activité et chaque quart doit respecter des contraintes dont certaines relèvent de la réglementation de travail et d'autres des règles de transfert. Ils commencent à des périodes spécifiques dans la journée. De plus chaque quart doit respecter les durées minimales et maximales de quart ainsi que les qualifications et les

jours de repos de l'employé. Durant son quart de travail, un employé peut être transféré au plus une fois. Selon la convention collective de travail, un employé a un nombre maximal d'heures de travail par semaine. Il est aussi important d'avoir un temps minimal de repos entre deux quarts consécutifs pour permettre aux employés d'être en forme et de ne pas souffrir de problèmes physiques ou mentaux dûs à la surcharge du travail.

L'objectif du problème est de sélectionner un sous-ensemble de quarts pour chaque employé pour satisfaire au mieux la courbe de demande de chaque département tout en minimisant les coûts totaux. Les coûts sont décrits ci-bas. Les quarts de travail peuvent se composer d'un ou deux blocs de travail. Un bloc de travail est défini comme un ensemble de périodes consécutives dans l'horizon de planification. Pour caractériser un bloc de travail, nous spécifions son début, sa durée et le département où se déroule le travail.

Selon la demande et le nombre d'employés disponibles, il se peut qu'il y ait une sous-couverture quand le nombre d'employés affectés est inférieur à celui requis pour une période i ou une sur-couverture quand le nombre d'employés affectés est supérieur à celui requis. Soit par exemple, deux employés notés e_1 et e_2 , et deux départements notés d_1 et d_2 . Les employés e_1 et e_2 ont le département d_1 pour département d'origine et sont qualifiés pour travailler au département d_2 . Les courbes de demande des deux départements sont présentées dans la figure 3.2.

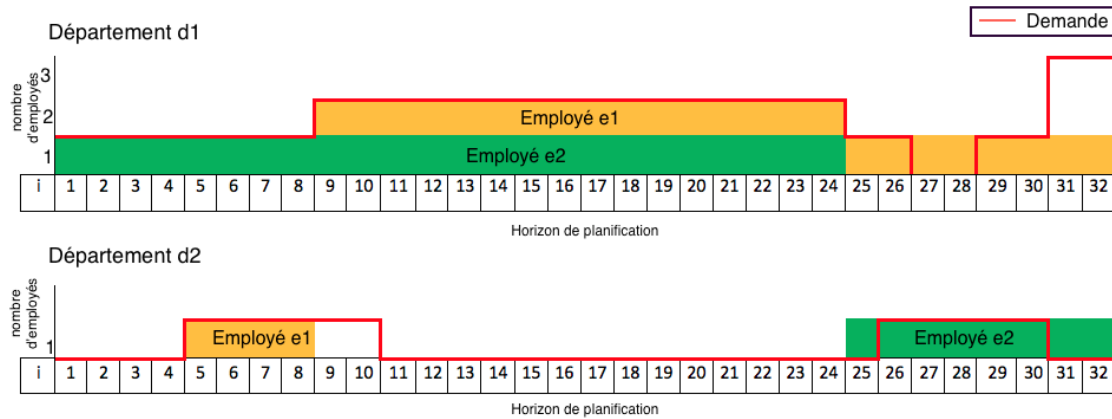


Figure 3.2 Exemple d'horaire avec deux employés et deux départements

Dans l'exemple de la figure 3.2, deux quarts sont retenus. L'employé e_1 commence son travail dans le département secondaire d_2 pour une durée d'une heure. Son deuxième bloc se déroule dans son département d'origine d_1 . Quant à l'employé e_2 , il commence son travail dans son département d'origine d_1 pour six heures. Ensuite, il est transféré à son département secondaire d_2 pour deux heures. Nous pouvons remarquer qu'il y a de la sous-couverture au

département d_1 aux périodes 31 et 32 et au département d_2 aux périodes 9 et 10. Il y a aussi de la sur-couverture au département d_1 aux périodes 27 et 28 et au département d_2 aux périodes 25, 31 et 32. Ces périodes de sous-couvertures et de sur-couvertures entraînent des coûts. Une sous-couverture est pénalisée par un coefficient c_u^d pour chacun des départements tandis qu'une sur-couverture est pénalisée par c_o^d pour chacun des départements. Dans le contexte multi-départements avec possibilité de transfert, des pénalités sont aussi utilisées pour réduire les transferts. Ces pénalités sont proportionnelles à la durée du bloc de transfert.

3.2 Génération de quarts

Les quarts sont générés selon les patrons admissibles et selon les jours de travail de l'employé. Un patron admissible est un patron dont la durée est admissible et dont le début est admissible. Chaque employé peut travailler au maximum dans deux départements pendant une journée dont l'un est son département d'origine. Il existe donc des patrons pour les quarts avec un seul bloc de travail et des patrons pour les quarts avec deux blocs de travail. L'employé peut travailler dans le département d'origine ou dans l'un de ses départements secondaires pendant toute la durée du quart ou travailler quelques périodes dans le département d'origine et quelques périodes dans l'un de ses départements secondaires. Dans le cas où le quart se compose de deux blocs de travail contigus, l'employé peut commencer par travailler dans le département d'origine ensuite terminer le quart dans un département de transfert ou par travailler dans un département de transfert ensuite terminer le quart dans le département d'origine.

Soit l_{min} la durée minimale d'un quart et l_{max} la durée maximale d'un quart. Nous supposons que chaque employé est payé pour les heures travaillées. La durée minimale d'un quart assure à l'employé une rémunération minimale s'il est appelé à travailler pendant un jour donné. La convention collective de travail restreint les quarts de travail très longs d'où le besoin d'une durée maximale autorisée par quart de travail.

Pour chaque employé $e \in E$, nous pouvons définir l'ensemble de quarts admissibles Q_e . Soit $q \in Q_e$ un quart admissible. Le quart q peut être constitué d'un seul bloc de travail ou de deux blocs de travail. Les quarts avec un seul bloc de travail sont les quarts internes quand l'employé travaille uniquement dans son département d'origine ou externes quand l'employé est transféré pour toute la durée du quart. Les quarts qui contiennent deux blocs de travail sont les quarts mixtes. En général, un quart peut être caractérisé par :

- L'employé assigné ;
- Le département d'origine de l'employé (s'il y a lieu) ;
- Le département de transfert (s'il y a lieu) ;

- La période de début de travail ;
- La durée de travail dans le département d'origine (s'il y a lieu) ;
- La durée de travail dans le département de transfert (s'il y a lieu).

Dans le cadre de notre projet, nous considèrerons une durée minimale d'une heure pour chaque bloc de travail pour les quarts mixtes.

Comme notre projet se veut une continuité du projet de Munezero (2014), nous allons utiliser les mêmes notations relatives à un quart de travail que nous présentons au tableau 3.1.

Nous pouvons caractériser un quart interne par le triplet (s_q, h_q, d_q) où d_q représente le département d'origine d_e de l'employé e . Si q est un quart externe pour l'employé e effectué dans un de ses départements secondaires $d_q \in D_e$, nous pouvons caractériser le quart q par (s_q, h_q, d_q) avec $d_q \neq d_e$. Si q est un quart mixte avec deux blocs de travail alors nous le caractérisons par le quintuplet $(s_q, h_q^b, h_q^f, d_q^b, d_q^f)$. Un des départements d_q^b où d_q^f doit être le département d'origine d_e de l'employé e . Afin de simplifier la notation, nous allons utiliser le quintuplet $(s_q, h_q^b, h_q^f, d_q^b, d_q^f)$ pour les trois types de quart dans le reste du mémoire. Dans le cas où le quart se compose d'un seul bloc de travail, nous allons mettre h_q^f à 0 et utiliser un département fictif pour d_q^f .

Soit Q_e^j l'ensemble de quarts admissibles de l'employé e au jour j . Le quart $q \in Q_e^j$ est dit admissible durant le jour j si et seulement s'il commence à une période dans le jour j , i.e., $s_q \in I_j$ et l'employé n'est pas en congé le jour j , i.e., $j \in J_e$.

Un coût de travail c_q est associé à chaque quart $q \in Q_e$. Ce coût dépend du salaire horaire c_e de l'employé e et de la durée du quart q . Le salaire horaire c_e peut être égal pour tous les employés mais peut aussi varier selon leurs anciennetés, leurs qualifications et leurs expertises. Si le quart q est mixte ou externe, une pénalité est associée au bloc de transfert. Cette pénalité est proportionnelle à la durée du bloc de transfert et au coût de transfert c_e^t . Le coût de transfert n'est pas un coût réel mais permet de diminuer les transferts.

Tableau 3.1 Notations relatives à un quart de travail

Notation	Explication
s_q	La période de début du quart q
h_q	Durée du quart q
h_q^b	Durée du bloc de début
h_q^f	Durée du bloc de fin
d_q	Département de réalisation du quart
d_q^b	Département de réalisation du bloc de début
d_q^f	Département de réalisation du bloc de fin

La génération de quarts se fait explicitement en fonction des qualifications de chaque employé ainsi que de ses disponibilités. Cette modélisation s’inspire du modèle de Dantzig (1954) et le généralise au cas multi-départements. Dans la section suivante, nous détaillerons le modèle mathématique qui permettra de choisir le sous-ensemble optimal parmi l’ensemble des quarts générés.

3.3 Modèle global

Le modèle global de Munezero (2014) correspond à un programme en nombres entiers et permet de générer un horaire personnalisé des employés dans un contexte multi-départements en permettant le partage de la main-d’œuvre qualifiée entre les départements. Ce modèle permet d’avoir un horaire optimal quand il est possible de le résoudre à l’optimalité. Le but du modèle est de minimiser la somme des coûts de travail ainsi que les pénalités de transfert, de sous-couverture et de sur couverture pour chaque département durant l’horizon de planification tout en respectant les règles de la convention collective de travail et les règles de transfert.

3.3.1 Notation

En se basant sur les concepts définis dans la section 3.1, nous commençons par définir les ensembles, les paramètres, les constantes ainsi que les variables du problème. Nous passons ensuite à la formulation en nombres entiers du modèle global.

Ensembles :

- D : L’ensemble des départements ;
- E : L’ensemble des employés ;
- E_d : L’ensemble des employés internes et externes du département d ;
- I : L’ensemble des périodes de l’horizon de planification ;
- I_j : L’ensemble des périodes $i \in I$ pour le jour j ;
- J_e : L’ensemble des jours où l’employé e est disponible pour travailler ;
- Q_e^j : L’ensemble des quarts possibles de l’employé e pour le jour j .

Paramètres et constantes :

Commençons par définir les paramètres d’ordre général :

- L_{max} : La charge horaire maximale d'un employé par semaine ;
 R_{min} : La durée de repos minimale entre deux quarts successifs ;
 δ^t : Le coefficient de pénalité de transfert ;
 δ^u : Le coefficient de pénalité de sous-couverture ;
 δ^o : Le coefficient de pénalité de sur-couverture.

Voici les paramètres associés aux départements :

- r_d^i : La demande dans le département d à la période i ;
 c_d^u : Le coût d'une sous-couverture dans le département d ;
 c_d^o : Le coût d'une sur-couverture dans le département d .

Les employés n'ont pas tous les mêmes qualifications, donc les paramètres peuvent varier d'un employé à l'autre. Ce sont principalement :

- c_e : Le coût horaire de travail de l'employé e dans le département d'origine ;
 c_e^t : Le coût horaire supplémentaire si l'employé e est transféré.

Finalement, pour chaque quart nous définissons ci-dessous un ensemble de constantes :

- h_q : La durée totale du quart q
 h_q^t : La durée du bloc de transfert du quart q
 f_q^i : vaut 1 si le quart q finit à la période i
 v_q^i : vaut 1 si le quart q contient la période i mais ne finit pas à la période i
 θ_{dq}^i : vaut 1 si à la période i le bloc de travail du quart q se passe dans le département d

Variables de décision du problème :

Dans notre modèle, nous considérons deux types de variables. Les variables binaires X_q permettent de sélectionner les quarts. Les variables entières U_d^i et O_d^i permettent de calculer les sous-couvertures et les sur-couvertures par rapport à la courbe de demande de chaque département et au nombre d'employés affectés par période de l'horizon de planification.

$$X_q = \begin{cases} 1 : & \text{si le quart } q \text{ est choisi dans la solution} \\ 0 : & \text{sinon} \end{cases}$$

U_d^i : Le nombre d'employés en sous-couverture dans le département d à la période i .

O_d^i : Le nombre d'employés en sur-couverture dans le département d à la période i .

3.3.2 Formulation mathématique

Le modèle global pour le problème de planification d'horaires du personnel dans un contexte multi-départements peut se formuler comme le programme en nombres entiers suivant :

$$\min \sum_{e \in E} \sum_{j \in J_e} \sum_{q \in Q_e^j} (c_e h_q + \delta^t c_e^t h_q^t) X_q + \sum_{d \in D} \sum_{i \in I} (\delta^u c_d^u U_d^i + \delta^o c_d^o O_d^i) \quad (3.1)$$

s.c.

$$\sum_{e \in E_d} \sum_{j \in J_e} \sum_{q \in Q_e^j} \theta_{dq}^i X_q + U_d^i - O_d^i = r_d^i \quad \forall d \in D; i \in I \quad (3.2)$$

$$\sum_{q \in Q_e^j} X_q \leq 1 \quad \forall e \in E; j \in J_e \quad (3.3)$$

$$\sum_{j \in J_e} \sum_{q \in Q_e^j} h_q X_q \leq L_{max} \quad \forall e \in E \quad (3.4)$$

$$\sum_{q \in Q_e^j} f_q^i X_q + \sum_{q \in Q_e^{j+1}} \sum_{i' \in [i, i+R_{min}]} v_q^{i'} X_q \leq 1 \quad \forall e \in E; \{j, j+1\} \in J_e; i \in I_j \quad (3.5)$$

$$X_q \in \{0, 1\} \quad \forall e \in E; j \in J_e; q \in Q_e^j \quad (3.6)$$

$$U_d^i \geq 0 \quad \forall d \in D; i \in I \quad (3.7)$$

$$O_d^i \geq 0 \quad \forall d \in D; i \in I \quad (3.8)$$

Le modèle global (3.1)-(3.8) traite tous les départements à la fois. Tous les quarts admissibles sont énumérés. Nous rappelons que la solution retenue doit satisfaire les règles de la convention collective de travail et les règles de transfert. Parmi ces règles, certaines se modélisent dans les contraintes (3.2)-(3.5) et certaines sont prises en compte dans la génération de quarts admissibles comme le fait qu'il faut avoir au plus deux blocs de travail dans un quart et dans le cas où il y a deux blocs de travail, il faut que l'un d'eux se passe dans le département d'origine et aussi le fait d'avoir des durées minimales et maximales pour les blocs de travail.

L'objectif (3.1) est de minimiser une somme pondérée des coûts : les coûts de travail, les coûts de transfert, les coûts de sous-couverture et les coûts de sur-couverture. Les contraintes (3.2) imposent la couverture de la demande et les ajustements se font à travers les variables U_d^i en cas de manque d'employés et les variables O_d^i en cas de surplus d'employés pour une période $i \in I$ donnée. En effet, pour chaque période de l'horizon de planification, le nombre d'employés requis est égal au nombre d'employés affectés plus la valeur de la sous-couverture engendrée moins la valeur de la sur-couverture engendrée. Les contraintes (3.3) assurent qu'un employé travaille au plus un seul quart par jour quand il n'est pas en congé. De plus un employé ne peut travailler plus que L_{max} heures durant l'horizon de planification d'après les contraintes

(3.4). Les contraintes (3.5) imposent un repos minimal de R_{min} périodes entre deux quarts affectés pour deux jours consécutifs. En d'autres mots, si un quart de travail qui commence à la période $i \in I$ du jour $j \in J$ est retenu dans la solution, nous interdisons l'affectation d'un quart au jour $j + 1 \in J$ qui contient une période i' dans l'intervalle $[i; i + R_{min}]$.

Les contraintes (3.6)-(3.8) assurent l'intégralité et la non négativité des variables.

3.4 Méthode de résolution du modèle global

Il est possible de résoudre le modèle global par n'importe quel optimiseur de programmation linéaire mixte (MIP). Nous utilisons l'optimiseur Xpress-MP de FICO pour résoudre le modèle en nombres entiers (3.1)-(3.8).

Comme vu dans la revue de littérature, les problèmes deviennent impossible à résoudre quand la taille augmente. Il est assez difficile voir même impossible de résoudre les problèmes de grande taille dans des temps de calcul raisonnables.

Dans le chapitre suivant, nous allons détailler une approche heuristique de résolution du modèle global en deux phases. Cette heuristique permet de diminuer considérablement les temps de calcul.

CHAPITRE 4 MÉTHODE DE RÉOLUTION EN DEUX PHASES

Dans ce chapitre, nous commençons par introduire le modèle de la première phase dans la section 4.1. Ensuite, nous présentons la deuxième phase à la section 4.2. Ce mémoire se concentre sur la deuxième phase et nous proposons différentes approches pour générer les quarts avec transfert.

4.1 Première phase : résolution département par département

La première phase est une résolution département par département. Pour chaque département, nous considérons que seuls les employés internes peuvent répondre à la demande. Aucun transfert n'est pris en compte. Cela nous permet d'avoir un sous-problème par département qui peut être résolu indépendamment des autres départements. Cependant, il se peut que les employés internes d'un département ne soient pas capables de répondre à la demande. La solution obtenue en première phase peut avoir plus de sous-couverture que la solution du modèle global d'où le besoin de la phase de ré-optimisation que nous étudions à la section 4.2. Une fois que chacun des sous-problèmes est résolu, nous obtenons un horaire optimisé avec seulement des quarts internes que nous utilisons comme point de départ pour l'énumération des quarts avec transfert.

4.1.1 Énoncé du problème mono-département

L'heuristique en deux phases résout le même problème exposé dans le chapitre précédent. Pour la première phase, la définition du problème de planification d'horaires du personnel de la section 3.1 est restreinte à un seul département $d \in D$.

4.1.2 Génération des quarts internes

Pour cette première phase, nous considérons les quarts à un seul bloc de travail. Pour chaque employé, nous générons tous les quarts internes possibles. Chaque quart est caractérisé par le triplet (s_q, h_q, d_q) où s_q représente la période de début du quart, h_q la durée du quart et d_q correspond au département d'origine d_e de l'employé $e \in E_d^{int}$.

Le nombre de quarts est nettement inférieur à celui des quarts générés pour le modèle global. Les quarts internes respectent les règles de construction de quarts comme la durée minimale l_{min} et la durée maximale l_{max} .

4.1.3 Formulation mathématique du modèle de la première phase

Nous utilisons la formulation de la section 3.3.2 restreinte à un département $d \in D$ à la fois en utilisant l'ensemble E_d^{int} comme ensemble d'employés. Ce modèle permet de générer un horaire personnalisé pour les employés E_d^{int} dans un contexte mono-département. Parfois, cette formulation ne donne pas la solution la plus adaptée pour créer les transferts optimaux.

Récemment Attia (2016) propose une meilleure approche qui permet de résoudre le modèle de la première phase d'une manière plus efficace pour avoir des quarts internes mieux adaptés aux transferts dans la deuxième phase. Cette approche se déroule en trois étapes. L'idée est de faire une prévision sur la sous-couverture de la première phase dans la première étape, de prévoir des transferts dans la deuxième étape et dans la troisième étape de résoudre le problème de planification d'horaires du personnel pour chaque département en prenant en compte les informations fournies par les deux étapes précédentes. En d'autres termes, avant de fournir tous les quarts possibles pour le modèle mathématique de la première phase, deux étapes préliminaires sont faites. Lors de ces étapes, des quarts anonymes sont générés pour couvrir la demande. A la première étape, des quarts anonymes internes permettent d'avoir une idée sur la couverture de la demande pour chaque département. A la deuxième étape, des quarts anonymes avec transferts permettent de savoir quels sont les départements qui peuvent fournir leurs employés internes pour les autres départements en sous-couverture. Finalement, la troisième étape est la résolution du problème de planification d'horaires du personnel pour les employés E_d^{int} où des quarts internes et des quarts avec les possibilités de transfert retenues à l'étape précédente sont énumérés. Cette méthode en trois étapes est plus efficace que la première phase classique où uniquement les quarts internes sont générés. Les résultats préliminaires de l'heuristique en deux phases incluant la nouvelle approche de résolution de la première phase seront présentés dans le chapitre 5.

4.2 Deuxième phase : ré-optimisation en permettant les transferts

Il est important d'avoir une deuxième phase pour avoir une vue d'ensemble sur tous les départements. Le but est de construire un horaire global incluant les quarts internes, externes et mixtes. L'étape la plus importante pour pouvoir atteindre la solution optimale est la génération des quarts avec transfert qui sera étudiée dans la section 4.2.2. Le modèle mathématique et la méthode de résolution sont assez semblables à ceux du modèle global.

La deuxième phase se veut être une phase de ré-optimisation ayant pour but d'améliorer la solution obtenue dans la première phase en ajoutant les transferts des employés entre les départements. La résolution se fait simultanément pour tous les départements. Dans ce

mémoire, trois approches ont été proposées.

4.2.1 Énoncé du problème de ré-optimisation

Soit une entreprise avec un ensemble de départements D , un ensemble d'employées E et un ensemble de quarts internes Q_e pour chaque employé $e \in E$. Cet ensemble de quarts retenus dans la première phase constitue un horaire admissible initial. Nous souhaitons l'utiliser pour créer des quarts avec transfert afin de diminuer la sous-couverture et la sur-couverture de la solution obtenue en première phase. Pour chaque employé $e \in E_d^{ext}$ qui peut être transféré occasionnellement au département $d \in D$, nous générons un nouvel ensemble de quarts Q_e dont fait partie les quarts internes retenus dans la solution obtenue en première phase, les quarts mixtes et les quarts externes. Nous rappelons qu'un quart mixte pour un employé $e \in E_d^{ext}$ est un quart qui se compose de deux blocs de travail contigus. Un des blocs se passe dans le département d'origine d_e de l'employé e et un autre bloc se passe dans l'un de ses départements secondaires $d \in D_e$. Un quart externe quant à lui est effectué en totalité dans l'un des départements secondaires $d \in D_e$ de l'employé $e \in E$. Notre but est de choisir simultanément le sous-ensemble optimal des quarts pour tous les employés afin de satisfaire la demande de tous les départements en utilisant les notions de sous-couverture et de sur-couverture définies précédemment. L'horaire global doit satisfaire la réglementation de la convention collective et les règles de transfert. Les règles de transfert exigent qu'un employé soit transféré au plus une seule fois par jour pour un seul de ses départements secondaires. Cela se traduit dans la génération de quarts avec transfert à la section prochaine. Il est aussi important d'avoir une durée minimale de travail soit dans le département d'origine ou dans le département de transfert. Si un employé qui travaille déjà dans son département d'origine se voit transférer à un autre département, ses durées de travail dans son département d'origine et dans son département de transfert doivent être supérieures ou égales à une heure. Cela évite d'avoir des blocs de travail trop courts. Dans la section 4.2.2, nous présentons les différentes approches pour transformer un quart interne en un quart de transfert admissible.

4.2.2 Génération des quarts avec transfert

Dans cette section, nous présentons trois approches de génération de quarts avec transfert. L'amélioration s'est faite de façon incrémentale. Nous avons commencé par développer une approche qui permet de créer des quarts mixtes en ajoutant un bloc de transfert au début ou à la fin d'un quart interne sans le modifier. Nous nous sommes rendus compte que cette façon de faire ne nous permet pas d'avoir la solution globale optimale résultante du modèle global. Pour espérer atteindre cette solution globale optimale, il faudrait changer les quarts internes.

C'est ce qui fait l'objet de la deuxième approche. Finalement, vu que nous nous basons sur la solution obtenue en première phase pour générer les quarts mixtes, il arrive parfois que cette solution ne soit pas adaptée pour atteindre la solution optimale. Dans ce cas, nous proposons de l'améliorer tout en restant réalisable. Dans la suite, nous allons détailler les trois approches que nous avons proposées.

Première approche : Pas de remise en compte des quarts internes

Comme nous l'avons cité précédemment, notre première approche permet de générer des quarts de transfert sans remettre en question les quarts obtenus dans la première phase. Pour chaque employé $e \in E_d^{ext}$, des quarts avec transfert sont générés durant les périodes où il y a de la sous-couverture au département $d \in D$. Les quarts externes sont générés quand un employé $e \in E_d^{ext}$ n'a aucun quart interne attribué pour un jour donné. Les quarts mixtes dépendent des quarts internes retenus dans la solution de la première phase. Chaque bloc de travail est caractérisé par son début, sa durée et le département où il se passe. Pour cette première approche, nous considérons qu'un bloc de travail interne concorde avec un quart interne de la solution obtenue en première phase. En outre, comme notre but est de diminuer le nombre de sous-couvertures obtenu de la première phase, un quart mixte n'est généré que dans le cas où son bloc de transfert au département $d \in D$ permet de couvrir au moins une sous-couverture dans ce département.

Formulation des quarts avec transfert : Généralisons maintenant les quarts avec transfert que cette approche permet de générer. Soit Q_e^j l'ensemble de quarts internes retenus dans la solution de la première phase pour chaque employé $e \in E$ au jour $j \in J_e$. Soient l_{min} la durée minimale d'un quart de travail, l_{max} la durée maximale d'un quart de travail, l_{min}^b la durée minimale d'un bloc de travail et $S_t \in I$ l'ensemble des débuts admissibles pour un quart. Soit un quart interne $q \in Q_e^j$ caractérisé par le triplet (s_q, h_q, d_q) où s_q représente la période de début du quart, h_q la durée en nombre de périodes du quart et d_q le département d'origine de l'employé e . Et soit, I_u^d l'ensemble des périodes en sous-couverture au département $d \in D_e$ qui est un département de transfert de l'employé e . Nous voulons créer un quart mixte avec un transfert de d_q vers d qui pourrait couvrir directement au moins une période $i \in I_u^d \cap I_j$ avec I_j l'ensemble des périodes au jour j . Soit q_t , un quart avec transfert. Si q_t est un quart mixte avec deux blocs de travail caractérisé par le quintuplet $(s_{q_t}, h_{q_t}^b, h_{q_t}^f, d_{q_t}^b, d_{q_t}^f)$ où s_{q_t} représente la période de début du quart q_t , $h_{q_t}^b$ la durée en nombre de périodes du bloc de début, $h_{q_t}^f$ la durée en nombre de périodes du bloc de fin, $d_{q_t}^b$ le département de réalisation du bloc de début et $d_{q_t}^f$ le département de réalisation du bloc de fin. Le quart mixte q_t peut commencer

ou finir avec le bloc interne qui correspond au quart q . Le quart q_t qui commence avec le bloc interne est admissible si et seulement si

$$q_t = (s_{q_t}, h_{q_t}^b, h_{q_t}^f, d_{q_t}^b, d_{q_t}^f) : \begin{cases} s_{q_t} = s_q \\ h_{q_t}^b = h_q \\ d_{q_t}^b = d_q \\ h_{q_t}^f \in [l_{min}^b, l_{max} - h_q] \mid \exists i \in [s_{q_t} + h_{q_t}^b, s_{q_t} + h_{q_t}^b + h_{q_t}^f] \cap I_u^d \\ d_{q_t}^f = d \end{cases}$$

Si le quart q_t finit avec le bloc interne, il est admissible si et seulement si

$$q_t = (s_{q_t}, h_{q_t}^b, h_{q_t}^f, d_{q_t}^b, d_{q_t}^f) : \begin{cases} s_{q_t} = (s_q - h_{q_t}^b) \in S_t \\ h_{q_t}^b \in [l_{min}^b, l_{max} - h_q] \mid \exists i \in [s_{q_t} - h_{q_t}^b, s_{q_t} - 1] \cap I_u^d \\ d_{q_t}^b = d \\ h_{q_t}^f = h_q \\ d_{q_t}^f = d_q \end{cases}$$

Si $Q_e^j = \emptyset$ alors des quarts externes sont générés. Un tel quart externe q_t est admissible si :

$$q_t = (s_{q_t}, h_{q_t}^b, h_{q_t}^f, d_{q_t}^b, d_{q_t}^f) : \begin{cases} s_{q_t} \in S_t \\ h_{q_t}^b \in [l_{min}^b, l_{max}] \mid \exists i \in [s_{q_t}, s_{q_t} + h_{q_t}^b - 1] \cap I_u^d \\ d_{q_t}^b = d \\ h_{q_t}^f = 0 \\ d_{q_t}^f = d_{fictif} \end{cases}$$

Illustrons cela par un exemple. Soit une instance avec deux départements d_1 et d_2 et quatre employés e_1, e_2, e_3 et e_4 . L'instance est notée par $IA(|D|, |E|)$ avec $|D|$ le nombre de départements et $|E|$ le nombre d'employés.

Le tableau 4.1 présente les caractéristiques et les règles de construction des quarts à respecter pour ce premier exemple. La figure 4.1, quant à elle, présente la courbe de demande pour chaque département de l'instance.

Comme pour le modèle global, nous générons d'une manière explicite l'ensemble de tous les quarts admissibles.

Quatre quarts internes ont été retenus pour la première phase et sont présentés dans le tableau 4.2.

Tableau 4.1 Caractéristiques de l'instance IA(2,4)

Les employés		
	Département d'origine	Département secondaire
e_1	d_2	d_1
e_2	d_1	d_2
e_3	d_2	d_1
e_4	d_1	d_2
L'horizon de planification : 1 jour		
Règles de construction des quarts de travail		
Durée minimale d'un quart		6h
Durée maximale d'un quart		10h
Durée minimale d'un bloc de travail		1h
Les débuts admissibles (toutes les 15min)	les quarts du matin	entre 6h ($i = 25$) et 9h ($i = 37$)
	les quarts de l'après-midi	entre 12h ($i = 49$) et 14h ($i = 57$)
	les quarts du soir	entre 21h ($i = 85$) et 22h ($i = 89$)

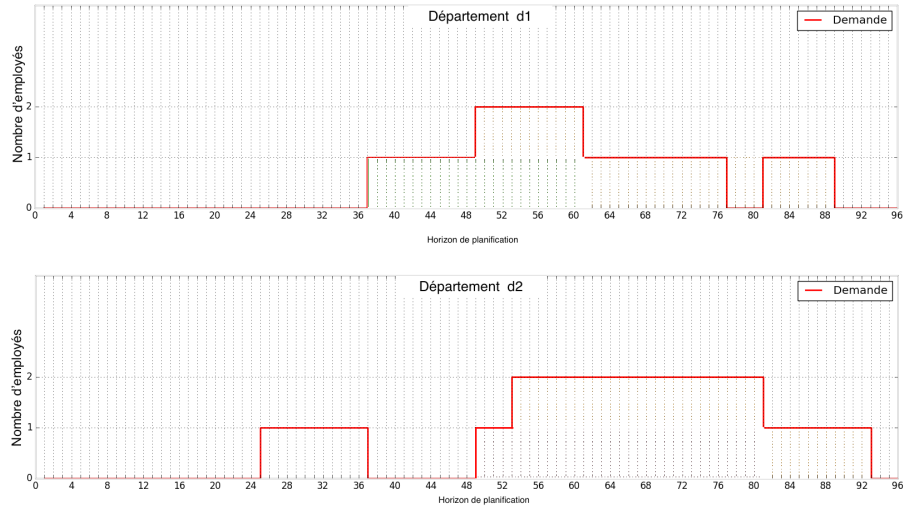


Figure 4.1 Courbes de demande des deux départements de l'instance IA(2,4) sur une journée

La figure 4.2 présente la solution du modèle de la première phase pour chaque département où chaque couleur correspond à un quart d'un employé. Nous pouvons remarquer qu'il y a de la sur-couverture dans le département d_1 entre la période $i = 77$ et la période $i = 80$ et de la sous-couverture dans le département d_2 entre la période $i = 25$ et la période $i = 36$. Pour couvrir la sous-couverture au département d_2 , regardons d'abord les employés transférables à ce département. Dans notre cas, seuls les employés e_2 et e_4 peuvent être transférés au département d_2 . Les blocs de transfert générés peuvent être ajoutés au début ou à la fin d'un

Tableau 4.2 Les quarts internes retenus pour l'instance IA(2,4)

	Période de début	Durée	Département
e_1	49	8h	d_2
e_2	49	9h	d_1
e_3	53	10h	d_2
e_4	37	6h	d_1

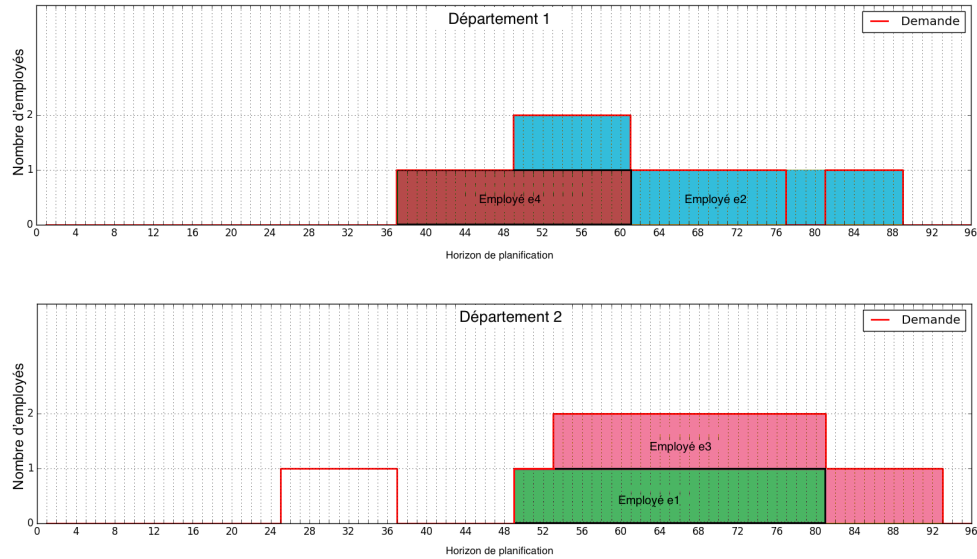


Figure 4.2 Solution de la première phase pour l'instance IA(2,4)

quart interne. Les quarts mixtes ainsi générés doivent respecter les règles de construction des quarts de travail énoncées dans le tableau 4.1. Dans le tableau 4.3, nous présentons la totalité des quarts avec transfert admissibles. Dans cet exemple, tous les employés de l'instance ont déjà des quarts internes attribués donc aucun quart externe ne sera généré. Les quarts avec transfert seront des quarts mixtes à deux blocs de travail.

Tableau 4.3 Les quarts de transfert possibles pour la première approche du modèle de ré-optimisation

		Bloc Interne				Bloc externe			
		Période de début	Période de Fin	Durée	Département	Période de début	Période de fin	Durée	Département
		Le quart commence avec le bloc interne							
q_1	e_2	49	88	9h	d_1	89	93	1h	d_2
q_2	e_4	37	60	6h	d_1	61	64	1h	d_2
q_3	e_4	37	60	6h	d_1	61	68	2h	d_2
q_4	e_4	37	60	6h	d_1	61	72	3h	d_2
q_5	e_4	37	60	6h	d_1	61	76	4h	d_2
		Le quart finit avec le bloc interne							
q_6	e_4	37	60	6h	d_1	33	36	1h	d_2
q_7	e_4	37	60	6h	d_1	29	36	2h	d_2
q_8	e_4	37	60	6h	d_1	25	36	3h	d_2

Pour ne pas générer les variables qui ne serviront pas, il est important de ne retenir que les quarts qui permettent d'améliorer la solution obtenue en première phase. Pour cette approche, seules les quarts qui peuvent couvrir directement une sous-couverture nous intéressent. Dans cet exemple, il est possible de couvrir le reste de la demande du département d_2 seulement avec les quarts q_6, q_7, q_8 . Et c'est le quart q_8 qui sera choisi dans la solution de ré-optimisation en utilisant la première approche puisque c'est le seul qui peut couvrir la totalité de la sous-couverture au département d_2 .

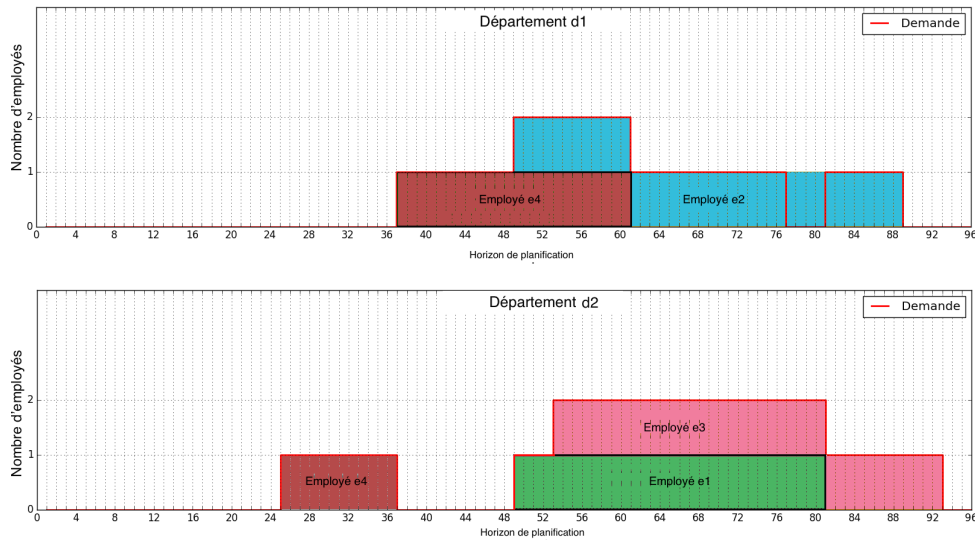


Figure 4.3 Solution pour l'instance IA(2,4) en utilisant la première approche de la phase de ré-optimisation

Les quarts avec transfert de cette première approche présentent l'inconvénient que ce sont des quarts qui ont été allongés. Ceci crée donc des journées de travail exigeantes car elles sont longues en plus de se dérouler dans deux départements. Cette solution peut être améliorée s'il était possible de modifier les quarts retenus dans la première phase. C'est ce qui fera l'objet du prochain paragraphe.

Deuxième approche : remise en question des quarts internes de la première phase

La deuxième approche permet d'ajouter plus de flexibilité à la phase de ré-optimisation et cela en remettant en question les quarts internes résultant de la première phase. La génération des quarts mixtes dépend de la sous-couverture et de la sur-couverture issue de la solution obtenue en première phase. Si un quart engendre de la sur-couverture dans la première phase, il peut être raccourci tout en restant admissible. Dans l'exemple précédent, nous pouvons remarquer que le quart interne de l'employé e_2 génère une sur-couverture au département d_1 entre la période $i = 77$ et la période $i = 80$. Pour enlever cette sur-couverture, il faudrait que le quart interne de l'employé e_2 finisse à la période $i = 76$ et qu'un bloc de transfert de 2h soit ajouté pour couvrir la demande entre la période $i = 81$ et la période $i = 88$. Dans cet exemple, l'employé e_1 est transférable au département d_1 et finit son quart interne à la période $i = 80$. La solution est présentée dans la figure 4.4.

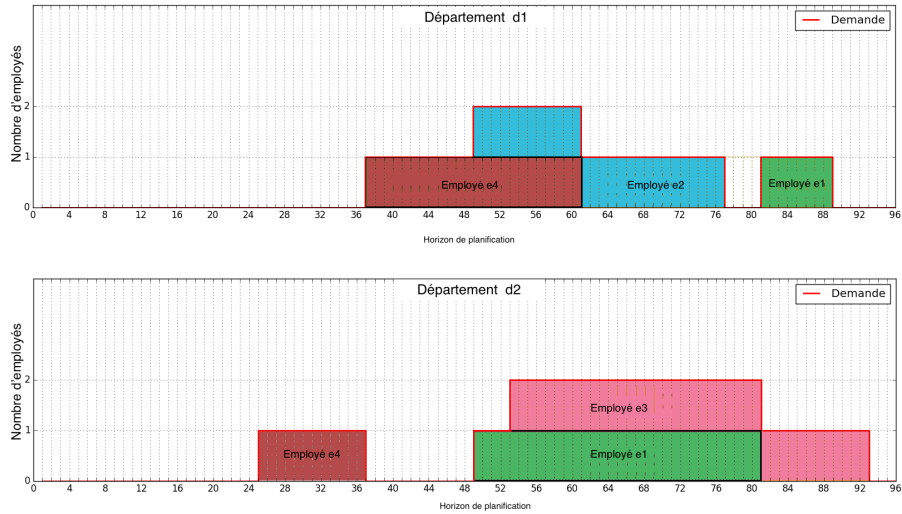


Figure 4.4 Solution pour l'instance IA(2,4) en utilisant la deuxième approche de la phase de ré-optimisation

Sur cela, nous présentons les différentes manières de modifier un quart interne et les possibilités d'en générer un quart mixte.

Stratégies de construction d'un quart mixte à partir d'un quart interne : Nous proposons de raccourcir ou d'allonger un quart interne au début ou à la fin et d'ajouter un bloc de transfert d'un côté ou d'un autre du bloc interne.

Dans l'exemple des figures 4.3 et 4.4, nous avons raccourci le quart interne de l'employé e_2 sans lui ajouter un bloc de transfert et cela parce qu'il contribuait dans la création d'une sur-couverture dans la solution obtenue en première phase. Ensuite, nous avons ajouté un transfert pour l'employé e_1 . Cela a permis d'améliorer la solution globale.

Il se peut que de la sur-couverture soit créée au début ou à la fin d'un quart interne et cela à cause des règles de construction de quarts. Par exemple, si nous avons une demande d'une durée de 5h et que la durée minimale d'un quart est de 6h, nous savons que tous les quarts générés vont créer une sur-couverture inévitable de 1h ou encore nous pouvons avoir une demande qui commence à 10h et que les débuts admissibles des quarts de travail sont entre 6h et 9h. Dans ce cas, chaque quart retenu créera une sur-couverture inévitable de 1h. Il se peut aussi que les mêmes périodes de sur-couverture dans un département $d_1 \in D$ correspondent à des sous-couvertures dans un autre département $d_2 \in D$. Dans ce cas, nous pouvons transformer le quart interne qui crée de la sur-couverture dans d_1 en un quart mixte avec un transfert vers d_2 tout en respectant les règles de construction des quarts et les règles de transfert.

Il est possible que de la sur-couverture soit créée au milieu d'un quart interne. Dans ce cas, en enlevant un bout du quart interne de l'employé $e \in E$, nous pouvons créer une nouvelle sous-couverture dans le département d'origine $d_e \in D$ de l'employé e . Pour satisfaire ce manque d'employé généré, nous pouvons étendre un autre quart interne ou transférer un employé vers le département d_e . Le but est de combler la nouvelle sous-couverture.

Lors de chacune des transformations, il faut tenir compte des durées et débuts admissibles du quart généré. En effet, un quart doit toujours respecter les règles de la convention collective et les règles de transfert. Différents quarts avec transfert sont suggérés mais c'est au modèle mathématique proposé dans la prochaine section de choisir la meilleure solution globale admissible.

Troisième approche : Permutation des quarts internes de la première phase

La troisième approche vient améliorer la deuxième approche. En plus de remettre en question les quarts de la première phase, il est possible que la solution de la première phase ne soit pas optimale, c'est-à-dire que les quarts des employés ne soient pas bien répartis dans la journée pour permettre le transfert. En voila un exemple explicatif. Soit une instance IA(2,3)

avec deux départements d_1 et d_2 et trois employés e_1 , e_2 et e_3 . Le tableau 4.4 présente les caractéristiques de cette instance.

Tableau 4.4 Caractéristiques de l'instance IA(2,3)

Les employés		
Employé	Département d'origine	Département secondaire
e_1	d_1	
e_2	d_1	d_2
e_3	d_2	d_1

La figure 4.5 présente les courbes de demande à couvrir pour les deux départements de l'instance IA(2,3).

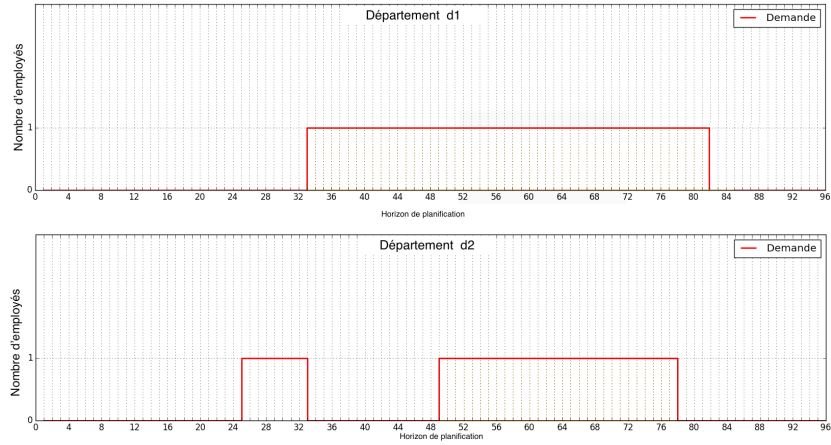


Figure 4.5 Courbes de demande des deux départements de l'instance IA(2,3) pour une journée

Trois quarts internes ont été retenus pour la première phase de l'instance IA(2,3) et sont présentés dans le tableau 4.5.

Tableau 4.5 Les quarts internes retenus pour l'instance IA(2,3)

Employé	Type du quart	Période de début	Durée	Département
e_1	interne	33	6h	d_1
e_2	interne	57	6h	d_1
e_3	interne	49	6h	d_2

Dans la figure 4.6, nous pouvons remarquer qu'aucun transfert n'a été retenu après la ré-optimisation en permettant la modification des quarts internes (deuxième approche). La raison est que les quarts internes retenus ne le permettent pas. Pour l'instance IA(2,3), nous

pouvons remarquer dans la figure 4.6 que l'employé e_1 est le seul disponible pour couvrir la sous-couverture au département d_2 entre la période $i = 25$ et $i = 32$ mais qu'il n'est pas transférable au département d_2 .

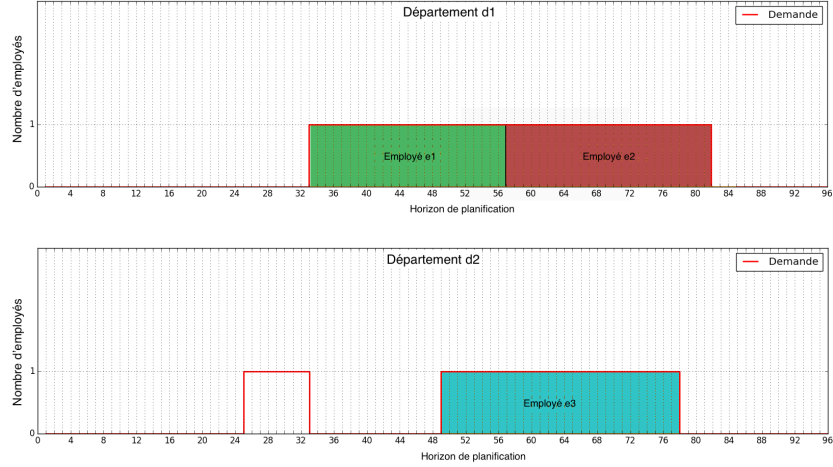


Figure 4.6 Solution de la phase de ré-optimisation avec la deuxième approche pour l'instance IA(2,3)

L'objectif est de trouver un autre employé dont le département d'origine est le même que l'employé e_1 , qu'il soit disponible à travailler en ce jour et qu'il puisse être transférable au département d_2 . Dans cet exemple, l'employé e_2 est un bon candidat. Nous proposons de modifier l'horaire initial de la première phase en permutant les quarts internes de l'employé e_1 et de l'employé e_2 . Ensuite nous proposerons un ensemble de quarts mixtes pour le nouveau quart interne de l'employé e_2 en utilisant la même manière de générer des quarts mixtes à partir d'un quart interne de la deuxième approche de l'heuristique en deux phases.

La figure 4.7 présente la solution de la ré-optimisation en utilisant la troisième approche pour l'instance IA(2,3).

Après avoir généré les quarts qui permettront d'améliorer la solution obtenue en première phase, il faudrait choisir la meilleure combinaison. Le modèle mathématique de la section 4.2.3 permet de choisir le sous-ensemble optimal parmi les quarts générés.

Dans le tableau 5.4 du chapitre suivant, les différences entre le nombre de quarts générés pour le modèle global et l'heuristique en deux phases avec les trois approches seront présentées. Le but du prochain chapitre est de montrer l'efficacité de l'heuristique en deux phases en terme de rapidité tout en gardant une bonne qualité de la solution.

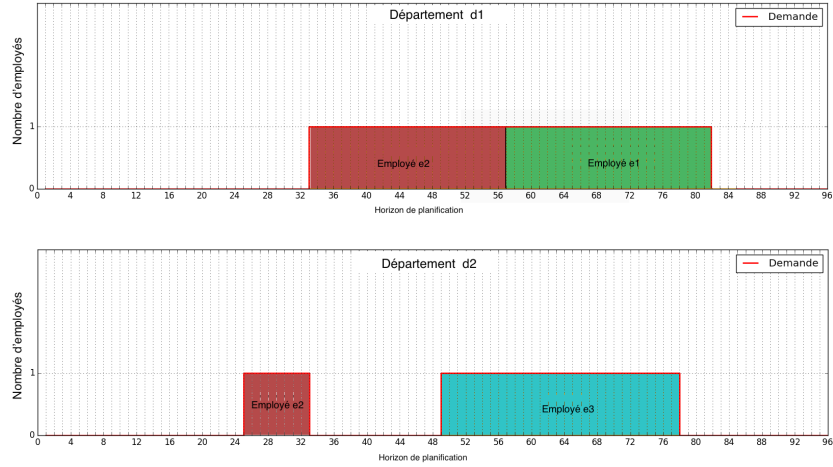


Figure 4.7 Solution de la phase de ré-optimisation avec la troisième approche pour l'instance IA(2,3)

4.2.3 Formulation mathématique

Dans la phase de ré-optimisation, nous utilisons le modèle mathématique en nombres entiers proposé dans la section 3.3.2. Le modèle global (3.1)-(3.8) traite tous les départements à la fois. L'ensemble des quarts rassemble les quarts internes retenus dans la solution obtenue en première phase ainsi que les quarts internes modifiés et les quarts avec transfert (mixtes et externes) selon chacune des approches proposées. L'objectif est de choisir une solution globale qui satisfait la demande de tous les départements à la fois en minimisant les coûts de travail, les coûts de transfert, les coûts de sous-couverture et les coûts de sur-couverture. Comme pour le modèle global, il faut respecter la réglementation concernant :

- la charge total de travail par employé qui est de L_{max} ;
- les deux jours de repos consécutifs pour chaque employé $e \in E$;
- la durée minimale d'un quart l_{min} ;
- la durée maximale d'un quart l_{max} ;
- le repos minimal entre deux quarts consécutifs qui est de R_{min} ;
- l'interdiction d'avoir plus d'un quart de travail par jour par employé.

Dans le chapitre suivant, les résultats numériques seront discutés. Les tests ont été réalisés sur un ensemble d'instances générées aléatoirement qui simulent des problèmes réels.

CHAPITRE 5 ANALYSE DES RÉSULTATS

Afin de valider l’heuristique en deux phases décrite dans le chapitre précédent, nous examinons les résultats obtenus pour les trois approches proposées en les comparant aux résultats du modèle global. D’abord nous présentons, à la section 5.1, l’environnement de test, les instances ainsi que les paramètres du problème. Ensuite, comme notre méthode consiste à générer explicitement l’ensemble total des quarts admissibles, il est important d’avoir un ordre de grandeur sur le nombre de quarts générés pour chaque modèle. A la section 5.2, nous exposons le nombre de quarts générés pour le modèle global et l’heuristique en deux phases. Pour qualifier la solution obtenue par l’heuristique en deux phases, nous avons établi certains critères d’évaluation. Dans ce cadre, nous observons, d’une part, les temps de résolution du problème pour chaque approche proposée. Et d’autre part, Nous comparons, le coût de la solution de l’heuristique en deux phases avec le coût de la solution optimale du modèle global. Il sera important aussi de connaître le taux de satisfaction de la demande par rapport à la demande totale, le nombre de quarts et le nombre de transferts. C’est ce qui est présenté dans la section 5.3.

Finalement, nous analysons les résultats des différents tests effectués tout au long du projet. Les résultats du modèle global sont détaillés à la section 5.4 et ceux de l’heuristique en deux phases à la section 5.5. A la dernière section de ce chapitre, nous présentons les améliorations faites par Attia (2016) sur la première phase de l’heuristique en deux phases et les résultats obtenus après la ré-optimisation.

5.1 Environnement de test

Pour valider les modèles en nombre entiers présentés précédemment, nous avons utilisé l’optimiseur commercial Fico Xpress-MP. Le code a été développé en Java et les modèles implémentés avec la bibliothèque Xpress-BCL (*Builder Component Library*).

L’ordinateur utilisé a Oracle Linux Server comme système d’exploitation et Intel (R) Core (TM) i7 comme processeur avec une fréquence d’horloge de 2,8 GHz et une mémoire de 128 Go.

5.1.1 Présentation des instances de test

Tout au long du projet, nous avons eu accès à un ensemble de jeux de données. Les données ont été générées aléatoirement par Dahmen (2013) mais reproduisent des problèmes réels.

Nous pouvons répartir ces données en un ensemble d'instances de petite taille et un ensemble d'instances de grande taille. Il est possible de résoudre le modèle global à l'optimalité pour la première catégorie d'instances et comparer la solution de l'heuristique en deux phases avec la solution globale optimale. Pour les grandes instances, la taille ne permet pas toujours la résolution en nombres entiers du modèle global à l'optimalité.

Nous présentons dans cette section les caractéristiques des instances de test qui nous ont été transmises. Chaque instance est caractérisée par un ensemble de départements D et un ensemble d'employés E . Et

Nous avons fixé les coûts comme suit :

$$\begin{aligned} c_d^u &= 5 \text{ pour tous les départements;} \\ c_d^o &= 2,5 \text{ pour tous les départements;} \\ c_e &= 10; \\ c_e^t &= 2. \end{aligned}$$

Les instances sont réparties en un ensemble de classes distinctes. Chaque classe est caractérisée par le nombre de départements et le nombre d'employés. Trois catégories de départements sont citées par Dahmen (2013) : les grands départements, les départements de taille moyenne et les petits départements. Trois niveaux de qualification sont retenus pour les employés. Il y a des employés hautement qualifiés, des employés moyennement qualifiés et des employés peu qualifiés. Chaque employé est affecté à un ensemble de départements en prenant en compte son niveau de qualification et la taille des départements. Chaque classe est notée par $(|D|, |E|)$ avec $|D|$ le nombre de départements et $|E|$ le nombre d'employés

D'après Dahmen (2013), on définit différents profils de demande, pour chacune des classes qui correspondent à l'ensemble des courbes de demande des départements de l'instance. En construisant une courbe de demande pour un département donné, on s'assure qu'il existe un horaire réalisable pour répondre à cette demande en prenant en compte les employés qualifiés à travailler dans ce département. Dans les cas réels, l'offre en nombre d'employés ne correspond pas toujours à la demande. Pour s'ajuster à la réalité, l'auteur introduit artificiellement des hausses et des baisses de demande.

Chaque instance est notée par $IA(|D|, |E|, p)$ avec $|D|$ le nombre de départements, $|E|$ le nombre d'employés et p le profil de demande. Nous obtenons douze différentes instances de petite taille que nous décrivons dans le tableau 5.1 et six instances de grande taille que nous décrivons dans le tableau 5.2

Tableau 5.1 Les instances de test de petite taille générées aléatoirement

Instance	Nombre de départements	Nombre d'employés	Profil
IA(2,20,1)	2	20	1
IA(2,20,4)	2	20	4
IA(2,50,1)	2	50	1
IA(2,50,3)	2	50	3
IA(3,20,1)	3	20	1
IA(3,20,2)	3	20	2
IA(3,50,1)	3	50	1
IA(3,50,3)	3	50	3
IA(3,70,1)	3	70	1
IA(5,20,1)	5	20	1
IA(5,20,3)	5	20	3
IA(5,50,1)	5	50	1

Tableau 5.2 Les grandes instances de test générées aléatoirement

Instance	Nombre de départements	Nombre d'employés	Profil
IA(5,70,1)	5	70	1
IA(5,70,2)	5	70	2
IA(5,200,1)	5	200	1
IA(10,200,1)	10	200	1
IA(10,300,1)	10	300	1
IA(10,400,1)	10	400	1

Comme nous l'avons déjà cité, les petites instances peuvent être résolues à l'optimalité avec le modèle global mais en des temps de résolution très grands. Par contre, il n'est pas possible de résoudre toutes les grandes instances à l'optimalité.

Quand la taille d'une instance est très grande, nous nous contentons de la résolution de la relaxation linéaire. La relaxation linéaire donne une idée sur la borne inférieure de la fonction objectif.

5.1.2 Valeurs des paramètres et constantes du problème

Pour l'ensemble des instances présentées dans les tableaux 5.1 et 5.2, l'horizon de planification est d'une semaine ($|J| = 7$) divisé en 672 périodes d'une durée de 15 minutes chacune. Pour les instances de petite taille ainsi que IA(5,70,1), $l_{min} = 6h$, $l_{max} = 10h$ et les durées varient entre l_{min} et l_{max} avec un pas d'une heure. Pour les instances de grande taille $l_{min} = 7h$, $l_{max} = 9h$ et le pas est de 30 minutes.

Chaque employé travaille au plus cinq jours par semaine. Il a deux jours de congés successifs pré-assignés dans un niveau décisionnel antérieur. La durée totale maximale des quarts assignés à un employé est $L_{max} = 40h$ par semaine et le repos minimal entre deux quarts affectés à un employé est $R_{min} = 12h$.

Pour minimiser les transferts, nous avons décidé de les pénaliser dans la fonction objectif du modèle mathématique présenté à la section 3.3 par un coefficient $\delta^t = 10$. Les sous-couvertures et les sur-couvertures sont aussi pénalisées respectivement par les coefficients $\delta^u = 100$ et $\delta^o = 100$.

5.2 Le nombre de quarts générés par instance

Le modèle global, s'il est résolu à l'optimalité, donne la meilleure solution au problème de planification d'horaires du personnel dans un contexte multi-départements avec possibilité de transfert entre les départements. Cependant, la résolution de ce modèle prend beaucoup de temps d'où le besoin de développer l'heuristique en deux phases dont fait objet ce mémoire. Le tableau 5.3 présente le nombre de quarts du modèle global générés pour les instances présentées dans les tableaux 5.1 et 5.2.

Pour le modèle global, nous pouvons remarquer dans le tableau 5.3 qu'il peut dépasser les 10^6 quarts. C'est ce qui explique la difficulté que nous avons observée pour résoudre ce modèle sur certaines instances et les temps de résolution.

Pour la première phase, chaque département est résolu séparément des autres. Pour chaque département $d \in D$, les quarts générés sont uniquement des quarts internes pour les employés E_{int}^d . Le tableau A.1, dans l'annexe A, présente le nombre de quarts internes générés par département pour toutes les instances de test. Nous pouvons remarquer dans le tableau 5.4 que le nombre de quarts générés pour la première phase est nettement inférieur à celui du modèle global. Nous nous attendons à ce que les temps de résolution diminuent considérablement par rapport au temps de résolution du modèle global.

Rappelons brièvement que, la première approche est celle qui génère le moins de quarts en deuxième phase. Elle est la plus restrictive puisque elle ne permet pas de remettre en question les quarts internes de la première phase. Les quarts de transfert avec un ou deux blocs de travail sont générés pour les jours où il y a de la sous-couverture. Le tableau A.2 présente le nombre de quarts générés par type pour cette première approche. Dans le tableau 5.4, nous pouvons observer que le nombre de quarts générés par la deuxième phase en utilisant la première approche est négligeable par rapport à celui du modèle global. Nous nous attendons à ce que les temps de résolution diminuent.

Tableau 5.3 Nombre de quarts générés pour le modèle global

Instance	Modèle Global			
	Quarts internes	Quarts mixtes	Quarts externes	Total
Instances de petite taille				
IA(2,20,1)	13 500	122 850	8 775	145 125
IA(2,20,4)	13 500	122 850	8 775	145 125
IA(2,50,1)	33 750	274 050	19 575	327 375
IA(2,50,3)	33 750	274 050	19 575	327 375
IA(3,20,1)	13 500	236 250	16 875	266 625
IA(3,20,2)	13 500	236 250	16 875	266 625
IA(3,50,1)	33 750	604 800	43 200	681 750
IA(3,50,3)	33 750	604 800	43 200	681 750
IA(3,70,1)	47 250	831 600	59 400	938 250
IA(5,20,1)	13 500	519 750	37 125	570 375
IA(5,20,3)	13 500	519 750	37 125	570 375
IA(5,50,1)	33 750	1 228 500	87 750	1 350 000
Instances de grande taille				
IA(5,70,1)	47 250	1 682 100	120 150	1 849 500
IA(5,70,2)	45 500	459 680	33 800	538 980
IA(5,200,1)	130 000	1 290 640	94 900	1 515 540
IA(10,200,1)	130 000	4 464 200	328 250	4 922 450
IA(10,300,1)	195 000	6 161 480	453 050	6 809 530
IA(10,400,1)	260 000	8 919 560	655 850	9 835 410

Le tableau A.3 présente le nombre de quarts générés pour cette deuxième approche. Dans certaines instances, le nombre de quarts pour cette deuxième approche est considérablement grand par rapport à celui de la première approche mais reste petit par rapport au modèle global. Cela s'observe sur les instances IA(2,50,1) et IA(3,70,1) ainsi que sur les instances de grande taille. La raison est que, il y a beaucoup de sous-couvertures qui restent à couvrir après la première phase. Nous observerons aussi un grand nombre de sous-couvertures avec la solution optimale du modèle global. Cette sous-couverture est de l'ordre de 15% pour les instances IA(2,50,1) , IA(3,70,1), IA(5,50,1) et IA(5,70,1).

Le tableau A.4 présente le nombre de quarts générés pour la troisième approche de la phase ré-optimisation. Quand il est favorable de permuter deux quarts internes, le nombre de quarts internes augmente comme observé dans le tableau A.4. Parfois ce nombre ne change pas par rapport à celui de la deuxième approche, donc la solution après ré-optimisation sera la même que celle de la deuxième approche. Ceci est le cas des instances IA(2,50,3), IA(3,20,2) et IA(5,20,3).

Tableau 5.4 Comparaison du nombre de quarts total entre le modèle global, la première phase et les trois approches de la phase de ré-optimisation

Instances de petite taille					
Instance	Modèle global	Première phase	Phase de ré-optimisation		
			1 ^{ère} approche	2 ^{ème} approche	3 ^{ème} approche
IA(2,20,1)	145 125	13 500	132	556	612
IA(2,20,4)	145 125	13 500	122	693	1 124
IA(2,50,1)	327 375	33 750	3 290	9 559	12 128
IA(2,50,3)	327 375	33 750	402	1 460	1 460
IA(3,20,1)	266 625	13 500	145	664	933
IA(3,20,2)	266 625	13 500	184	934	934
IA(3,50,1)	681 750	33 750	294	1 019	1 589
IA(3,50,3)	681 750	33 750	456	2 703	3 491
IA(3,70,1)	938 250	47 250	12 103	32 288	43 693
IA(5,20,1)	570 375	13 500	217	658	713
IA(5,20,3)	570 375	13 500	139	1 396	1 396
IA(5,50,1)	1 350 000	33 750	11 972	40 044	44 095
Instances de grande taille					
IA(5,70,1)	1 849 500	47 250	21 953	59 730	69 314
IA(5,70,2)	538 980	45 500	1 198	43 531	94 889
IA(5,200,1)	1 515 540	130 000	3 978	160 144	512 136
IA(10,200,1)	4 922 450	130 000	9 604	529 393	1 062 740
IA(10,300,1)	6 809 530	195 000	20 355	731 144	1 858 911
IA(10,400,1)	9 835 410	260 000	14 005	1 097 015	2 954 388

Dans la suite du chapitre, nous exposons les résultats numériques obtenus pour les instances présentées dans les tableaux 5.1 et 5.2. D'abord, dans la section 5.3, nous introduisons les critères qui nous permettent d'évaluer les résultats numériques obtenus et qualifier la qualité de la solution. Puis, nous présentons, les résultats obtenus avec le modèle global à la section 5.4. Ensuite nous comparerons les différentes approches présentées dans ce mémoire dans la section 5.5. Finalement à section 5.6 de ce travail, nous présentons une amélioration de ce travail sur les deux phases principalement sur la première phase de l'heuristique proposée.

5.3 Les critères de performance

Pour analyser les résultats obtenus, nous observons d'abord les temps de résolution de chaque modèle. En considérant que la solution optimale est celle du modèle global, nous comparons le coût de cette solution par rapport aux coûts des solutions retenues par les approches développées pour l'heuristique en deux phases. Le coût étant la valeur de la fonction objectif pour chaque modèle qui est une somme des coûts de travail, des pénalités de transfert, des pénalités de sous-couverture et des pénalités de sur-couverture sur l'ensemble du problème

qu'il soit mono-département ou multi-départements. L'écart entre la valeur de la solution globale optimale et la valeur de chaque solution obtenue pour les deux phases de l'heuristique est noté *Gap*. Il est calculé en pourcentage par rapport à la valeur de la solution optimale. Pour chaque instance, nous déterminons, la demande totale, le nombre total de sous-couvertures et le nombre total de sur-couvertures pour tous les départements pendant toutes les périodes de l'horizon de planification I . Les formules (5.1)-(5.3) permettent de calculer, respectivement, la demande totale, la sous-couverture totale et la sur-couverture totale sur l'ensemble des périodes I pour tous les départements D .

$$\sum_{d \in D} \sum_{i \in I} r_d^i \quad (5.1)$$

$$\sum_{d \in D} \sum_{i \in I} U_d^i \quad (5.2)$$

$$\sum_{d \in D} \sum_{i \in I} O_d^i \quad (5.3)$$

D'autres critères peuvent être évalués comme le pourcentage de couverture de la demande calculée par la formule (5.4), le nombre de quarts retenus et le nombre de transferts effectués.

$$\frac{\sum_{d \in D} \sum_{i \in I} (r_d^i - U_d^i)}{\sum_{d \in D} \sum_{i \in I} (r_d^i)} \quad (5.4)$$

5.4 Résultats du modèle global

Toutes les instances de petite taille ont été résolues à l'optimalité avec le modèle global. Cela n'est pas toujours faisable pour les instances de grande taille. Les contraintes d'intégralité des cinq dernières instances du tableau 5.5 ont été relâchées. Nous estimons que la valeur de la relaxation linéaire donne une bonne indication sur la valeur de la solution entière optimale du modèle global car pour les petites instances les écarts entre les solutions de la relaxation linéaire et les solutions entières sont de l'ordre de 10^{-4} qui est la valeur d'arrêt du branchement par défaut de l'optimiseur.

Nous rapportons dans le tableau 5.5 la demande totale, la sous-couverture totale, la sur-couverture totale, le nombre de quarts retenus tous types confondus, le nombre de quarts avec transfert, la valeur de la fonction objectif, le temps de résolution en secondes ainsi que le taux de satisfaction de la demande pour la solution globale de chaque instance. Dans neuf des instances, la demande est satisfaite à plus de 99 %, ce qui veut dire qu'il est possible

d'avoir une solution sans manque d'employés. Dans quelques cas, il y a un surplus d'employés pour certaines périodes. Ceci compensera au niveau des opérations, le retard ou l'absence non prévu d'un employé. Dans ce cas le surplus peut donner de la flexibilité au responsable des ressources humaines pour combler ce manque d'employés. Parfois la demande est faite de façon irrégulière qu'il est impossible d'éviter les sous-couvertures et les sur-couvertures dans la solution optimale.

Tableau 5.5 Résultats numériques du modèle global pour toutes les instances

Instance	Demande	U^{finale}	O^{finale}	nb Quarts	nb Transferts	Objectif	Temps (sec)	Demande couverte (%)
Instances de petite taille								
IA(2,20,1)	3 140	2	2	100	2	9 520	31	99
IA(2,20,4)	3 052	2	7	98	4	11 459	34	99
IA(2,50,1)	7 347	835	100	204	30	460 770	129	88
IA(2,50,3)	7 604	5	5	243	11	23 590	87	99
IA(3,20,1)	2 968	1	5	99	1	9 580	53	99
IA(3,20,2)	2 956	3	3	99	5	10 250	53	99
IA(3,50,1)	7 564	0	8	245	5	21 600	250	100
IA(3,50,3)	7 632	2	2	246	4	21 160	253	99
IA(3,70,1)	10 292	1 480	131	279	34	797 180	384	85
IA(5,20,1)	3 112	0	12	98	9	11 230	668	100
IA(5,20,3)	3 048	4	4	99	2	10 900	119	99
IA(5,50,1)	7 448	997	96	203	70	542 700	909	86
Instances de grande taille								
IA(5,70,1)	10 164	1 496	137	278	55	807 120	821	85
IA(5,70,2)	10 938	Relaxation linéaire				191 090	4 560	—
IA(5,200,1)	31 267	Relaxation linéaire				538 987	50 520	—
IA(10,200,1)	31 167	Relaxation linéaire				636 488	60 340	—
IA(10,300,1)	46 700	Relaxation linéaire				860 562	40 650	—
IA(10,400,1)	62 254	Relaxation linéaire				1 058 806	55 840	—

5.5 Résultats de l'heuristique en deux phases

5.5.1 Résultats de la première phase

Rappelons que la première phase se déroule département par département et qu'il est possible de résoudre chaque sous-problème sur une machine indépendante d'une façon simultanée. La sous-couverture finale U^{finale} , la sur-couverture finale O^{finale} ainsi que la valeur de la fonction objectif du problème sont la somme de ces valeurs pour chaque département. Pour le temps de résolution, c'est le temps de résolution du département le plus long à résoudre qui est retenu.

Les temps de résolution du problème de planification d'horaires du personnel dans un contexte mono-département sont raisonnables pour les instances de petite taille. En général, ils dépassent rarement une minute. Pour les grandes instances, cela reste acceptable.

Tableau 5.6 Résultats numériques de la première phase de l'heuristique en deux phases

Instance	Demande	U ^{finale}	O ^{finale}	Objectif	GAP(%)	Temps (sec)
Instances de petite taille						
IA(2,20,1)	3 140	24	0	19 900	109	19
IA(2,20,4)	3 052	39	11	30 000	162	19
IA(2,50,1)	7 347	1 117	86	596 420	29	55
IA(2,50,3)	7 604	101	5	70 910	201	46
IA(3,20,1)	2 968	25	0	20 200	111	9
IA(3,20,2)	2 956	81	2	48 560	374	8
IA(3,50,1)	7 564	20	16	33 430	55	28
IA(3,50,3)	7 632	29	4	34 940	65	28
IA(3,70,1)	10 292	1 780	124	943 440	18	69
IA(5,20,1)	3 112	60	18	42 280	276	6
IA(5,20,3)	3 048	18	18	21 320	96	5
IA(5,50,1)	7 448	1 549	130	822 540	52	20
Instances de grande taille						
IA(5,70,1)	10 126	1793	125	949 790	19	14
IA(5,70,2)	10 938	1 127	305	665 210	248	43
IA(5,200,1)	31 267	3 169	798	1 856 400	244	440
IA(10,200,1)	31 167	3 068	873	1 818 740	185	414
IA(10,300,1)	46 700	5 350	1 270	3 086 785	258	65
IA(10,400,1)	62 254	5 530	1 701	3 336 760	215	474

D'après le tableau 5.6, il n'est pas possible d'atteindre la solution optimale uniquement avec la première phase. Pour les instances que nous avons testées, l'écart avec la solution optimale commence à 18% et peut atteindre plus de 300%. D'après les valeurs de sous-couverture, nous pouvons déduire qu'il y a forcément au moins un département qui n'arrive pas à couvrir d'une façon optimale sa demande avec uniquement ses employés internes. C'est pour cette raison que nous avons recours à une deuxième phase où nous introduisons les transferts.

5.5.2 Résultats de la phase de ré-optimisation

Les détails des résultats de la phase de ré-optimisation sont présentés en annexe B. Dans le tableau 5.7, nous comparons la valeur de la fonction objectif de la solution de l'heuristique en deux phases avec la solution du modèle global. Nous pouvons observer que effectivement, l'ajout de flexibilité à la phase de ré-optimisation, d'une approche à l'autre a permis de diminuer l'écart entre la valeur optimale du modèle de la deuxième phase et celle du modèle global. En d'autres termes, comme nous l'avons estimé précédemment, la qualité de la solution est améliorée, mais cela ne nous a pas permis d'atteindre exactement la solution optimale du

modèle global dans la majorité des instances. Cela est dû aux quarts retenus dans la première phase qui sont restrictifs pour la génération des quarts avec transfert.

Dans le tableau 5.8, nous pouvons observer l'augmentation des temps de résolution de la deuxième phase de l'heuristique en deux phases pour les instances testées d'une approche à l'autre.

Tableau 5.7 Résultats de la phase de ré-optimisation avec les différentes approches

Instance	Objectif du Modèle global	Phase de ré-optimisation					
		1 ^{ère} approche		2 ^{ème} approche		3 ^{ème} approche	
		Objectif	Gap (%)	Objectif	Gap (%)	Objectif	Gap (%)
Instances de petite taille							
IA(2,20,1)	9 520	9 520	0	9 520	0	9 520	0
IA(2,20,4)	11 459	26 530	131	24 360	112	17 100	49
IA(2,50,1)	460 770	510 930	10	495 160	7,4	487 159	5
IA(2,50,3)	23 590	67 920	187	51 149	116	51 150	116
IA(3,20,1)	9 580	16 130	68	15 930	66	15 930	66
IA(3,20,2)	10 250	30 140	194	24 060	134	24 060	134
IA(3,50,1)	21 600	29 210	35	28 460	31	25 480	18
IA(3,50,3)	21 160	33 120	56	33 080	56	31 370	48
IA(3,70,1)	797 180	860 420	8	830 160	4,1	828 100	3,8
IA(5,20,1)	11 230	27 580	145	17 470	55	17 470	55
IA(5,20,3)	10 900	21 320	95	12 400	13	12 400	13
IA(5,50,1)	542 700	613 130	12	576 089	6,1	574 110	5,7
Instances de grande taille							
IA(5,70,1)	807 120	852 270	5,5	820 649	1,6	818 560	1,4
IA(5,70,2)	191 090	583 954	205	476 350	149	461 760	141
IA(5,200,1)	538 987	1 617 669	200	1 349 070	150	1 056 954	96
IA(10,200,1)	636 488	1 531 650	140	1 221 930	92	1 036 895	63
IA(10,300,1)	860 562	2 370 589	175	1 569 364	82	1 549 009	80
IA(10,400,1)	1 058 806	2 866 840	170	1 991 589	88	1 958 702	84

Au tableau 5.8, nous pouvons observer les temps de résolution de l'heuristique et ceux du modèle global. Pour les instances de petite taille, les temps de résolution peuvent atteindre les 80 secondes. Les temps de résolution pour les trois approches sur les grandes instances sont aussi rapportés. Pour certaines instances, ces temps de calcul sont considérés très grands. La dernière section de ce chapitre présente une nouvelle approche pour résoudre ces instances en des temps raisonnables.

Tableau 5.8 Comparaison des temps de calcul

Instance	Temps de résolution (sec)			
	Modèle global	Première approche	Deuxième approche	Troisième approche
Instances de petite taille				
IA(2,20,1)	31	2	2	2
IA(2,20,4)	34	<1	<1	1
IA(2,50,1)	129	3	20	13
IA(2,50,3)	87	<1	1	1
IA(3,20,1)	53	<1	<1	<1
IA(3,20,2)	53	<1	<1	<1
IA(3,50,1)	250	<1	<1	<1
IA(3,50,3)	253	<1	<2	2
IA(3,70,1)	384	8	44	53
IA(5,20,1)	668	<1	<1	<1
IA(5,20,3)	119	<1	1	1
IA(5,50,1)	909	7	59	51
Instances de grande taille				
IA(5,70,1)	821	15	70	79
IA(5,70,2)	4 560 (RL)	3	35	3 649
IA(5,200,1)	50 520 (RL)	2	1 861	3 877
IA(10,200,1)	60 340 (RL)	9	11 234	23 213
IA(10,300,1)	40 650 (RL)	32	11 232	28 272
IA(10,400,1)	55 840 (RL)	31	14 300	27 560

5.5.3 Heuristique en deux phases avec la première phase optimale

Les résultats que nous venons de présenter à la section 5.5 montrent que la solution de la phase de ré-optimisation n'est pas optimale. Nous supposons que cela est dû aux quarts choisis dans la première phase. Pour valider cette hypothèse, nous avons fait le test avec une solution optimale de la première phase. Cette solution est déduite de la solution globale optimale et permet de fournir à la phase de ré-optimisation les meilleurs quarts internes possibles. Les quarts de la solution globale sont modifiés pour ne retenir que les quarts internes et les blocs internes pour les quarts mixtes. Si la durée d'un bloc interne est inférieur à la durée minimale d'un quart de travail l_{min} , un quart interne de durée minimale l_{min} est retenu. Les quarts externes sont éliminés. Les résultats après ré-optimisation en utilisant la deuxième et la troisième approches correspondent exactement à la solution optimale globale. Cela nous a permis de tester la validité de la deuxième phase.

A la prochaine section, nous avons fusionné le travail réalisé par Attia (2016) sur la première phase de l'heuristique avec notre projet. Cette collaboration a permis à l'heuristique en deux phases de produire de meilleurs résultats.

5.6 Amélioration de la première phase

Attia (2016) propose une amélioration de la première phase. La première phase classique génère tous les quarts internes pour chacun des départements sans considérer les transferts et la solution la moins coûteuse était choisie pour chaque département. Les mêmes valeurs des paramètres et constantes décrites à la section 5.1.2 sont prises en compte sauf pour l'ensemble des durées admissibles des quarts de travail pour les instances de petite taille et l'instance IA(5,70,1). Au lieu d'avoir des pas d'une heure, nous choisissons des pas de 15 minutes. Cette discrétisation permet d'avoir des quarts plus susceptibles de satisfaire la totalité de la demande. Les résultats du modèle global avec cette flexibilité sont présentés dans le tableau 5.9. Dans la nouvelle approche, la démarche se déroule en trois étapes résolues chacune par un modèle en nombre entiers. La première est la résolution du problème de planification d'horaires en utilisant des quarts anonymes et en respectant globalement les contraintes du problème. À l'issue de cette étape, une idée est faite sur les sous-couvertures qui résulteraient de la première phase. La deuxième étape est aussi une résolution du problème avec des quarts anonymes. Cette fois-ci, un modèle par jour est développé pour tous les départements à la fois. Cela permet d'identifier, par exemple, qu'un transfert du département $d_1 \in D$ vers le département $d_2 \in D$ peut combler de la sous-couverture dans ce dernier. La solution anonyme donne des quarts anonymes avec transfert qui servent à créer des quarts personnalisés à l'étape suivante. Finalement, la dernière étape permet de résoudre le problème de planification d'horaires personnalisés dans un contexte multi-départements en permettant le transfert des employés entre les départements. Deux approches de la première phase améliorée ont été proposées. Elles sont décrites dans les sections 5.6.1 et 5.6.2. La première donne une solution en des temps de calcul rapides et la deuxième donne une meilleure solution mais avec des temps de résolution plus longs. Il est à l'utilisateur de faire un arbitrage entre la qualité de la solution obtenue et le temps de résolution.

Si l'estimation à la première étape est bien faite, il est possible d'atteindre une solution optimale dans la première phase. La deuxième phase aura pour rôle d'améliorer la solution de cette première phase.

Tableau 5.9 Résultats numériques du modèle global pour les instances de petite taille

Instance	Demande	U ^{finale}	O ^{finale}	nb Quarts	nb Transferts	Objectif	Temps (sec)	Demande Couverte (%)
Instances de petite taille								
IA(2,20,1)	3 140	0	0	100	2	8 010	109	100
IA(2,20,4)	3 052	0	0	98	6	8 080	341	100
IA(2,50,1)	7 347	834	98	205	28	439 815	510	88
IA(2,50,3)	7 604	0	0	243	11	19 875	1 125	100
IA(3,20,1)	2 968	1	5	99	1	9 087	202	99,9
IA(3,20,2)	2 956	0	2	77	5	8 510	680	100
IA(3,50,1)	7 564	0	0	245	4	19 559	1 788	100
IA(3,50,3)	7 632	0	0	246	4	19 669	1 175	100
IA(3,70,1)	10 292	1 480	131	279	33	775 897	3 368	85
IA(5,20,1)	3 112	0	0	98	9	8 080	133	100
IA(5,20,3)	3 048	2	0	74	2	8 905	7 470	99,9
IA(5,50,1)	7 448	997	95	203	69	542 700	1 719	86
Instances de grande taille								
IA(5,70,1)	10 164	1 486	125	243	48	800 455	5 306	85

5.6.1 Nouvelle approche de la première phase : Temps de calcul rapide

Dans cette section, nous allons d'abord présenter la solution obtenue par cette nouvelle première phase. Ensuite, les résultats après ré-optimisation sont énoncés. Nous pouvons remarquer dans le tableau 5.10 qu'il est possible dans certains cas d'atteindre la solution optimale avec la nouvelle première phase. Pour nos instances de test, cela arrive pour deux instances.

Comme il y a des instances où nous n'arrivons pas à obtenir la solution optimale du modèle global du premier coup, il faudrait alors penser à une phase de ré-optimisation. D'abord, nous avons commencé par transformer la solution que nous obtenons après la première phase en une solution comportant uniquement des quarts internes comme nous l'avons fait pour transformer la solution du modèle global en une solution de la première phase dans la section 5.5.3. Puis, nous avons appliqué les trois approches de la phase de ré-optimisation mais nous nous sommes rendus compte qu'il vaudrait mieux prendre la solution de la première phase telle qu'elle est et ne pas la transformer. La raison est que la transformation en une solution de première phase et ensuite la régénération des quarts avec transfert ne fait que doubler le travail. Il est plus intéressant de regarder la solution avec ses sous-couvertures et ses sur-couvertures et essayer de l'améliorer. Pour ce faire, une adaptation de la phase de ré-optimisation est proposée. La transformation des quarts internes en des quarts mixtes expliquée dans le chapitre 4 est généralisée à tous les types de quarts. Nous pouvons donc raccourcir un quart, allonger un quart ou ajouter un bloc de transfert.

Dans le tableau 5.11, nous comparons les résultats des nouvelles solutions obtenues avec les nouveaux modèles des premières et des deuxième phases.

Tableau 5.10 Nouvelle approche de la première phase : Temps de calcul rapide

Instance	Demande	U ^{finale}	O ^{finale}	Objectif	Gap (%)	Temps (sec)
Instances de petite taille						
IA(2,20,1)	3 140	0	0	8 010	0	2
IA(2,20,4)	3 052	5	4	11 432	41	<1
IA(2,50,1)	7 347	842	99	463 885	5,4	3
IA(2,50,3)	7 604	29	48	46 027	131	13
IA(3,20,1)	2 968	0	5	9 087	0	6
IA(3,20,2)	2 956	2	2	9 505	11	<1
IA(3,50,1)	7 564	8	0	23 520	20	2
IA(3,50,3)	7 632	8	0	23 570	19	2
IA(3,70,1)	10 292	1 492	136	803 945	3,6	4
IA(5,20,1)	3 112	12	0	14 015	73	2
IA(5,20,3)	3 048	2	2	9 420	5,7	<1
IA(5,50,1)	7 448	1 162	140	634 657	16	16
Instances de grande taille						
IA(5,70,1)	10 164	1 500	152	812 757	3,1	4
IA(5,70,2)	10 938	495	439	387 070	102	3 601
IA(5,200,1)	31 267	823	1 070	766 610	42	3 601
IA(10,200,1)	31 167	893	1 101	808 710	27	3 602
IA(10,300,1)	46 700	5 907	1 400	3 423 862	297	3 604
IA(10,400,1)	62 254	6 301	2 741	3 998 675	277	4 023

Dans le tableau 5.11, nous pouvons observer la valeur objectif du modèle global ainsi que celle de la nouvelle première phase et la nouvelle deuxième phase. Après la deuxième phase, nous avons réussi à avoir des écarts de moins de 5 % pour plus de 75% des instances testées avec des temps de calcul raisonnables.

5.6.2 Nouvelle approche de la première phase : Temps de calcul allongé

Dans le but d'améliorer encore plus la solution de l'heuristique en deux phases, nous avons eu accès à des solutions de première phase qui ont été générées avec plus de flexibilité au niveau de l'étape 2. Nous rappelons que cette étape permet de créer des quarts anonymes qui peuvent être avec un ou deux blocs de travail et qui serviront à générer des quarts personnalisés. Cette approche génère beaucoup plus de quarts anonymes au niveau de la deuxième étape que celle que nous venons de présenter à la section 5.6.1. Pour cette raison, seules les instances de petite taille ont été résolues jusqu'à maintenant. Les résultats de la première phase qui nous ont été transmis sont présentés dans le tableau 5.12 et ceux de la deuxième phase dans le tableau 5.13.

Tableau 5.11 Résultats de la phase de ré-optimisation pour la nouvelle approche de la première phase (Temps de calcul rapide)

Instance	Objectif du modèle global	Première phase			Deuxième phase		
		Objectif	Gap(%)	Temps (sec)	Objectif	Gap (%)	Temps (sec)
Instances de petite taille							
IA(2,20,1)	8 010	8 010	0	2	8 010	0	1
IA(2,20,4)	8 080	11 432	41	<1	11 180	38	<1
IA(2,50,1)	439 815	463 885	5,4	3	455 807	3,6	40
IA(2,50,3)	19 875	46 027	131	13	44 040	121	<1
IA(3,20,1)	9 087	9 087	0	6	9 087	0	<1
IA(3,20,2)	8 510	9 505	11	<1	9 007	5,8	<1
IA(3,50,1)	19 559	23 520	20	2	19 580	0,1	<1
IA(3,50,3)	19 669	23 570	19	2	19 670	0	<1
IA(3,70,1)	775 897	803 945	3,6	4	797 180	2,7	63
IA(5,20,1)	8 080	14 015	73	2	12 045	49	<1
IA(5,20,3)	8 905	9 420	5,7	<1	9 420	5,7	1
IA(5,50,1)	542 700	634 657	16	16	559 874	3,1	72
Instances de grande taille							
IA(5,70,1)	788 215	812 757	3,1	4	800 985	1,6	107
IA(5,70,2)	191 090	387 070	102	3 601	320 499	67	338
IA(5,200,1)	538 987	766 610	42	3 601	735 807	36	5
IA(10,200,1)	636 488	808 710	27	3 602	786 889	23	633
IA(10,300,1)	860 562	3 423 862	297	3 604	2 652 632	208	9
IA(10,400,1)	1 058 806	3 998 675	277	4 023	3 580 730	238	25

Tableau 5.12 Nouvelle approche de la première phase : Temps de calcul allongé

Instance	Demande	U ^{finale}	O ^{finale}	Objectif	Gap (%)	Temps (sec)
IA(2,20,1)	3 140	0	0	8 010	0	2
IA(2,20,4)	3 052	5	4	11 432	41	<1
IA(2,50,1)	7 347	844	97	464 410	5,5	9
IA(2,50,3)	7 604	5	0	22 627	13	3
IA(3,20,1)	2 968	0	5	9 087	0	6
IA(3,20,2)	2 956	2	2	9 505	11	<1
IA(3,50,1)	7 564	8	0	23 520	20	2
IA(3,50,3)	7 632	8	0	23 570	19	2
IA(3,70,1)	10 292	1 481	131	798 412	2,9	20
IA(5,20,1)	3 112	12	0	14 015	73	2
IA(5,20,3)	3 048	2	2	9 420	5,7	<1
IA(5,50,1)	7 448	1 131	97	611 052	12,5	6
IA(5,70,1)	10 164	1 489	132	804 657	2	20

Dans le tableau 5.12, nous pouvons remarquer une amélioration au niveau de la première phase pour l'instance IA(2,50,3). En considérant les résultats des deux premières phases, nous pouvons remarquer que l'écart de la solution de la première phase par rapport à celle du

modèle global passe de 131% (tableau 5.10) à 13%(tableau 5.12). Ensuite, après la deuxième phase, l'écart est à 4% (tableau 5.13) au lieu de 121% (tableau 5.11).

Dans la figure 5.1, nous comparons les écarts des solutions obtenues dans la section 5.5 pour les trois approches de la deuxième phase avec la nouvelle approche. Les écarts calculés prennent pour solution optimale la solution du modèle global. Nous avons introduit plus de flexibilité pour la nouvelle génération de quarts dans le modèle global ainsi que pour les nouvelles premières et deuxième phases. La figure 5.1 permet de conclure que nous avons une meilleure solution de l'heuristique en deux phases avec les nouvelles premières et deuxième phases.

Tableau 5.13 Résultats de la phase de ré-optimisation pour la nouvelle approche de la première phase (Temps de calcul allongé)

Instance	Objectif du modèle global	Première phase			Deuxième phase		
		Objectif	Gap (%)	Temps (sec)	Objectif	Gap (%)	Temps (sec)
IA(2,20,1)	8 010	8 010	0	2	8 010	0	1
IA(2,20,4)	8 080	11 432	41	<1	11 180	38	<1
IA(2,50,1)	439 815	464 410	5,5	9	456 125	3,7	40
IA(2,50,3)	19 875	22 627	13	3	20 657	3,9	<1
IA(3,20,1)	9 087	9 087	0	6	9 087	0	<1
IA(3,20,2)	8 510	9 505	11	<1	8 510	0	<1
IA(3,50,1)	19 559	23 520	20	2	19 580	0,1	<1
IA(3,50,3)	19 669	23 570	19	2	19 670	0	<1
IA(3,70,1)	775 897	798 412	2,9	20	786 934	1,4	124
IA(5,20,1)	8 080	14 015	73	2	12 045	49	<1
IA(5,20,3)	8 905	9 420	5,7	<1	9 420	5,7	1
IA(5,50,1)	542 700	611 052	12	6	571 079	5,2	171
IA(5,70,1)	788 215	804 657	2	20	795 829	0,9	95

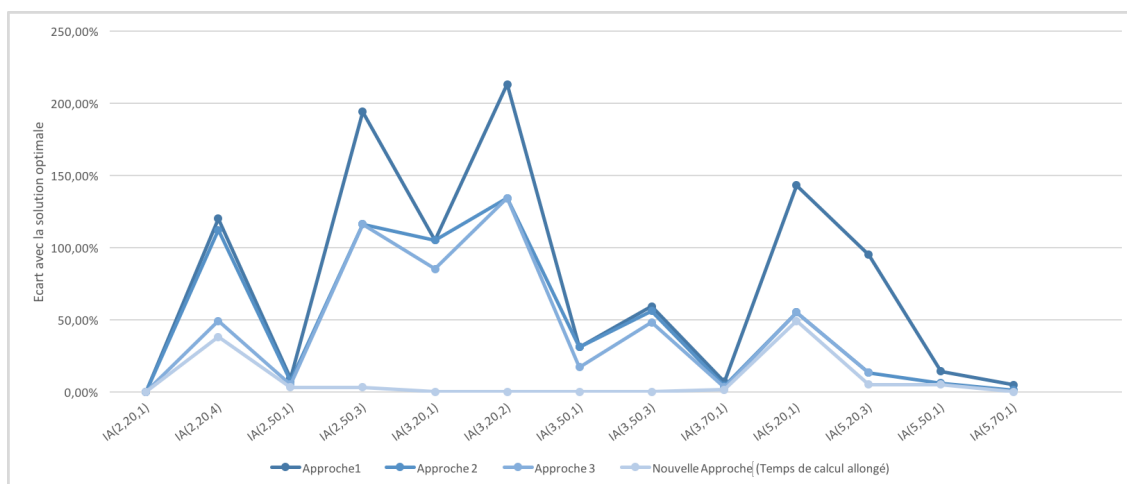


Figure 5.1 Comparaison des écarts des solutions de l'heuristique en deux phases classique et la nouvelle solution

Dans la figure 5.2, le temps de résolution de la dernière approche est supérieur à celui de la première approche que nous avons présentée au chapitre 4. Pour les instances avec 20 employés, il ne dépasse pas une seconde contrairement aux deuxième et troisième approches où le temps de résolution est plus long.

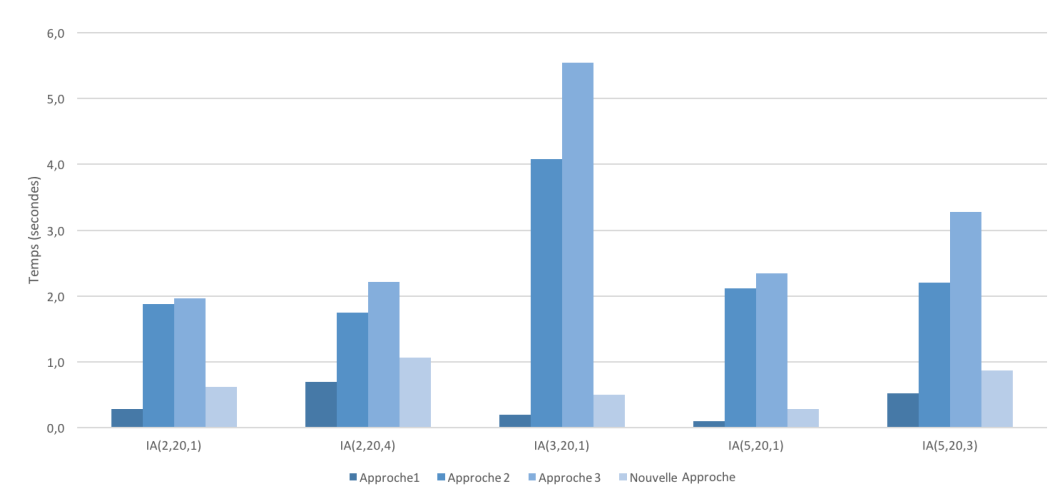


Figure 5.2 Comparaison des temps de résolution des solutions de l'heuristique en deux phases classique et la nouvelle solution

CHAPITRE 6 CONCLUSION

Dans ce projet, nous avons proposé une approche en deux phases pour résoudre le problème de planification d'horaires du personnel dans un contexte multi-départements avec la possibilité de partager la main-d'oeuvre. Nous nous sommes intéressés à la deuxième phase de l'heuristique. Étant donné une solution initiale retenue par le problème de planification d'horaires du personnel mono-département pour chaque département, nous introduisons la possibilité de transfert des employés entre les départements pour améliorer cette solution de première phase. Dans ce chapitre, nous concluons en synthétisant nos travaux à la section 6.1 et nous présentons les améliorations qui pourraient être apportées sur le projet à la section 6.2.

6.1 Synthèse des travaux

Dans ce mémoire, nous traitons le problème de planification d'horaires du personnel dans un contexte multi-départements avec la possibilité de transférer les employés entre les départements. Nous avons développé d'abord un modèle global qui permet de résoudre le problème en une seule étape et nous nous sommes rendus compte que ce modèle n'est pas efficace en terme de temps d'exécution. En effet, le nombre de quarts générés peut vite devenir très grand avec le nombre d'employés et le nombre de départements et il est parfois difficile de résoudre le problème voir même impossible pour les instances de grande taille. Pour résoudre ce problème de taille, nous avons appliqué une heuristique en deux phases. Cette heuristique consiste à résoudre le problème mono-département pour chaque département en première phase et ensuite à ré-optimiser la solution obtenue pour minimiser encore plus les sous-couvertures et les sur-couvertures créées. Nous avons commencé d'abord par développer une première phase classique où tous les quarts internes sont générés explicitement. La solution de moindre coût est choisie pour chacun des départements. Il se trouve que cette solution n'est pas optimale et requiert des améliorations. Ces améliorations se font en deuxième phase qui a une vue plus globale du problème. Nous avons développé trois approches pour améliorer la solution que nous avons obtenue en première phase. La première approche ne permet pas de modifier la solution initiale obtenue dans la première phase. L'ajout des transferts est très restreint. Pour la deuxième approche, nous nous permettons de modifier les quarts internes de la solution de la première phase. Un quart interne peut être allongé, raccourci en plus d'y rajouter des transferts. Finalement, la troisième approche permet de corriger les quarts internes de la première phase en permutant les employés entre eux quand cela est nécessaire pour ajouter des transferts. Avec ces trois approches, nous n'avons pas toujours obtenu une

solution optimale à l'issue de la phase de ré-optimisation. Nous avons alors émis l'hypothèse que c'était dû à la première phase qui nous empêchait de faire les transferts optimaux. Cela a été validé quand nous avons pris une solution de première phase optimale, qui n'est autre que la solution globale optimale restreinte aux blocs de travail internes et nous avons obtenu une solution finale optimale avec notre deuxième phase.

Vers la fin de ce projet, nous avons collaboré avec Attia (2016) qui travaille sur l'amélioration de la première phase de l'heuristique en deux phases. La première phase qu'elle propose se base sur la prédiction des sous-couvertures finales de la première phase et la prédiction des transferts possibles à l'aide de quarts anonymes pour ensuite générer des quarts mieux adaptés aux transferts en deuxième phase.

Les programmes en nombres entiers que nous avons proposés sont résolus avec l'optimiseur commercial FICO Xpress-MP. Pour les grandes instances, il n'était pas toujours possible de résoudre le modèle global à l'optimalité. Cependant, la solution du problème relaxé nous donne une idée sur la borne inférieure. Avec la nouvelle première phase, nous avons obtenu des solutions avec un écart d'optimalité inférieur à 5% pour 75% des instances et, cela, en des temps de calcul négligeables par rapport à ceux du modèle global.

6.2 Améliorations futures

Nous avons montré que l'amélioration de la première phase a permis d'améliorer la solution finale obtenue par l'heuristique en deux phases. Cependant les temps de résolution restent grands pour les instances de grande taille. En effet, plus nous ajoutons de quarts dans le but d'améliorer la solution, plus cela prend de temps pour résoudre le problème. Cela se remarque quand la taille des instances augmente. C'est à l'utilisateur de faire un arbitrage entre la qualité de la solution et les temps de résolution. Le travail se poursuit sur la première phase pour avoir une meilleure solution et pour en diminuer les temps de résolution. L'intérêt est porté sur les étapes de prédiction de la solution de la première phase.

De plus, nous avons effectué les tests sur des instances générées aléatoirement. Les tests sur des problèmes réels sont à envisager. Dans ce même contexte, nous pouvons adapter notre modèle pour prendre en compte les préférences des quarts de travail dans la journée. Il se peut que des employés préfèrent travailler pendant le jour ou pendant la nuit. Les qualifications et le niveau d'expérience pourraient aussi être pris en compte. Finalement, le travail s'est fait dans un contexte multi-départements où chaque département est mono-activité. Nous pouvons explorer les mêmes pistes de recherche dans un contexte multi-activités. Il faudrait alors adapter les règles de définition des quarts ainsi que le modèle mathématique.

RÉFÉRENCES

- I. Addou et F. Soumis, “Bechtold-Jacobs generalized model for shift scheduling with extraordinary overlap”, *Annals of Operations Research*, vol. 155, no. 1, pp. 177–205, 2007.
- H. K. Alfares, “Survey, categorization, and comparison of recent tour scheduling literature”, *Annals of Operations Research*, vol. 127, no. 1-4, pp. 145–175, 2004.
- D. Attia, “Optimization of employees shift scheduling with inter-departments transfers”, 2016. "Communication privée".
- K. R. Baker, “Workforce allocation in cyclical scheduling problems : A survey”, *Journal of the Operational Research Society*, vol. 27, no. 1, pp. 155–167, 1976.
- J. Bard et L. Wan, “Workforce design with movement restrictions between workstation groups”, *Manufacturing & Service Operations Management*, vol. 10, no. 1, pp. 24–42, 2008.
- S. E. Bechtold et L. W. Jacobs, “Implicit modeling of flexible break assignments in optimal shift scheduling”, *Management Science*, vol. 36, no. 11, pp. 1339–1351, 1990.
- , “The equivalence of general set-covering and implicit integer programming formulations for shift scheduling”, *Naval Research Logistics*, vol. 43, no. 2, pp. 233–249, 1996.
- P. Brucker, R. Qu, et E. Burke, “Personnel scheduling : Models and complexity”, *European Journal of Operational Research*, vol. 210, no. 3, pp. 467–473, 2011.
- M.-C. Côté, B. Gendron, et L.-M. Rousseau, “Modeling the regular constraint with integer programming”, dans *International Conference on Integration of Artificial Intelligence and Operations Research Techniques in Constraint Programming*. Springer, 2007, pp. 29–43.
- M.-C. Côté, B. Gendron, C.-G. Quimper, et L.-M. Rousseau, “Formal languages for integer programming modeling of shift scheduling problems”, *Constraints*, vol. 16, no. 1, pp. 54–76, 2011.
- M.-C. Côté, B. Gendron, et L.-M. Rousseau, “Grammar-based integer programming models for multiactivity shift scheduling”, *Management Science*, vol. 57, no. 1, pp. 151–163, 2011.
- S. Dahmen, “Construction de quarts de travail pour une main d’œuvre hétérogène sur des jours multiples et dans un contexte multi-départements”, 2013.

- G. B. Dantzig, "Letter to the editor-a comment on Edie's "traffic delays at toll booths"", *Journal of the Operations Research Society of America*, vol. 2, no. 3, pp. 339–341, 1954.
- L. C. Edie, "Traffic delays at toll booths", *Journal of the Operations Research Society of America*, vol. 2, no. 2, pp. 107–138, 1954.
- A. T. Ernst, H. Jiang, M. Krishnamoorthy, B. Owens, et D. Sier, "An annotated bibliography of personnel scheduling and rostering", *Annals of Operations Research*, vol. 127, no. 1-4, pp. 21–144, 2004.
- A. T. Ernst, H. Jiang, M. Krishnamoorthy, et D. Sier, "Staff scheduling and rostering : A review of applications, methods and models", *European Journal of Operational Research*, vol. 153, no. 1, pp. 3–27, 2004.
- L. W. Jacobs et S. E. Bechtold, "Labor utilization effects of labor scheduling flexibility alternatives in a tour scheduling environment", *Decision Sciences*, vol. 24, no. 1, pp. 148–166, 1993.
- H. C. Lau, "On the complexity of manpower shift scheduling", *Computers & Operations Research*, vol. 23, no. 1, pp. 93–102, 1996.
- E. Munezero, "Une heuristique en deux phases pour la confection d'horaires de personnel avec transferts inter-départementaux d'employés.", Mémoire de maitrise, Polytechnique Montréal, 2014.
- M. Rekik, J.-F. Cordeau, et F. Soumis, "Solution approaches to large shift scheduling problems", *RAIRO-Operations Research*, vol. 42, no. 2, pp. 229–258, 2008.
- , "Implicit shift Scheduling with multiple breaks and work stretch duration restrictions", *Journal of Scheduling*, vol. 13, no. 1, pp. 49–75, 2010.
- T. Aykin, "Optimal shift scheduling with multiple break windows", *Management Science*, vol. 42, no. 4, pp. 591–602, 1996.
- , "A composite branch and cut algorithm for optimal shift scheduling with multiple breaks and break windows", *Journal of the Operational Research Society*, vol. 49, no. 6, pp. 603–615, 1998.
- J. Van den Bergh, J. Beliën, P. De Bruecker, E. Demeulemeester, et L. De Boeck, "Personnel scheduling : A literature review", *European Journal of Operational Research*, vol. 226, no. 3, pp. 367–385, 2013.

ANNEXE A DÉTAILS SUR LE NOMBRE DE QUARTS GÉNÉRÉS

Tableau A.1 Nombre de quarts générés par instance pour le modèle de la première phase

Instance	Modèle de la phase séquentielle										
	Dep 1	Dep 2	Dep 3	Dep 4	Dep 5	Dep 6	Dep 7	Dep 8	Dep 9	Dep 10	Total
Instances de petite taille											
IA(2,20,1)	5 387	2 719									11 050
IA(2,20,4)	6 222	3 916									11 466
IA(2,50,1)	12 657	16 086									31 042
IA(2,50,3)	15 345	12 385									29 352
IA(3,20,1)	4 379	1 791	2 026								10 508
IA(3,20,2)	3 602	1 648	2 579								10 016
IA(3,50,1)	10 134	7 380	8 825								29 120
IA(3,50,3)	9 284	7 971	8 633								29 296
IA(3,70,1)	14 068	6 914	20 223								42 984
IA(5,20,1)	641	1 687	932	2 657	938						9 944
IA(5,20,3)	510	1 870	900	2 370	458						9 223
IA(5,50,1)	5 235	1 856	7 140	3 605	2 836						27 853
Instances de grande taille											
IA(5,70,1)	5 018	2 078	19 520	4 243	3 328						41 439
IA(5,70,2)	37 069	25 976	112	39 093	26 244						147 611
IA(5,200,1)	33 735	115 290	22 205	97 614	157 518						438 994
IA(10,200,1)	35 080	44 590	99 630	27 320	30 949	86 571	3 079	27 927	18 008	21 482	431 083
IA(10,300,1)	86 843	31 228	91 017	136 272	25 809	25 105	3 188	90 567	27 536	90 936	650 351
IA(10,400,1)	28 619	2 408	201 960	13 564	93 231	94 464	8 043	106 047	113 076	197 532	880 378

Tableau A.2 Nombre de quarts générés par instance pour le modèle de ré-optimisation sans remise en question des quarts internes

Instance	Modèle de ré-optimisation : 1 ^{ère} approche			Total
	Quarts internes	Quarts mixtes	Quarts externes	
Instances de petite taille				
IA(2,20,1)	100	32	0	132
IA(2,20,4)	98	32	0	130
IA(2,50,1)	197	197	2 943	3 337
IA(2,50,3)	239	24	27	290
IA(3,20,1)	98	20	0	118
IA(3,20,2)	96	34	27	157
IA(3,50,1)	246	42	0	288
IA(3,50,3)	244	98	135	477
IA(3,70,1)	272	813	10 962	12 047
IA(5,20,1)	98	65	54	217
IA(5,20,3)	99	40	0	139
IA(5,50,1)	190	1 145	10 665	12 000
IA(5,70,1)	264	1 544	22 113	23 921
Instances de grande taille				
IA(5,70,2)	314	334	546	1 194
IA(5,200,1)	872	826	4 238	5 936
IA(10,200,1)	893	3 368	6 916	11 177
IA(10,300,1)	1 317	4 060	11 908	17 285
IA(10,400,1)	1 772	6 264	18 018	26 054

Tableau A.3 Nombre de quarts générés par instance pour le modèle de ré-optimisation avec remise en question des quarts internes

Instance	Modèle de ré-optimisation : 2 ^{ème} approche		
	Quarts internes	Quarts mixtes	Quarts externes
Instances de petite taille			
IA(2,20,1)	100	1 629	0
IA(2,20,4)	98	1 645	54
IA(2,50,1)	197	19 003	2 268
IA(2,50,3)	239	3 390	81
IA(3,20,1)	98	2 021	0
IA(3,20,2)	96	2 029	54
IA(3,50,1)	246	2 765	0
IA(3,50,3)	244	7 832	0
IA(3,70,1)	272	62 847	9 801
IA(5,20,1)	99	1 999	0
IA(5,20,3)	99	4 633	0
IA(5,50,1)	190	83 001	12 636
Instances de grande taille			
IA(5,70,1)	264	114 666	19 710
IA(5,70,2)	318	34 918	612
IA(5,200,1)	872	136 789	8 313
IA(10,200,1)	893	813 180	22 338
IA(10,300,1)	1 317	602 669	11 908
IA(10,400,1)	1 351	746 404	18 018

Tableau A.4 Nombre de quarts générés par instance pour le modèle de ré-optimisation avec permutation d'employés

Instance	Modèle de ré-optimisation : 3 ^{ème} approche		
	Quarts internes	Quarts mixtes	Quarts externes
Instances de petite taille			
IA(2,20,1)	103	1 817	0
IA(2,20,4)	110	3 141	54
IA(2,50,1)	367	31 489	2 268
IA(2,50,3)	239	3 390	81
IA(3,20,1)	109	2 936	0
IA(3,20,2)	96	2 029	54
IA(3,50,1)	277	4 684	0
IA(3,50,3)	263	10 575	0
IA(3,70,1)	776	105 534	9 801
IA(5,20,1)	101	2 187	0
IA(5,20,3)	99	4 633	0
IA(5,50,1)	328	96 499	12 636
Instances de grande taille			
IA(5,70,1)	604	147 742	19 710
IA(5,70,2)	625	71 983	612
IA(5,200,1)	3 524	445 191	8 313
IA(10,200,1)	4 854	1 634 893	22 338
IA(10,300,1)	9 789	1 580 260	11 908
IA(10,400,1)	12 770	228 8747	18 018

ANNEXE B RÉSULTATS DE L'HEURISTIQUE EN DEUX PHASES

Tableau B.1 Résultats de la première phase de l'heuristique en deux phases

Instance	Id Dep	Demande	U^{finale}	O^{finale}	Objectif	Temps (sec)
IA(2,20,1)	1	1 894	16	0	12 760	19
	2	1 246	8	0	7 140	4
IA(2,20,4)	1	1 862	26	0	17 680	19
	2	1 190	13	11	12 320	11
IA(2,50,1)	1	3 315	429	45	233 370	38
	2	4 032	688	41	363 050	55
IA(2,50,3)	1	4 409	25	0	23 520	46
	2	3 195	76	5	47 390	37
IA(3,20,1)	1	1 403	12	0	9 480	9
	2	688	0	0	1 900	3
	3	877	13	0	8 820	3
IA(3,20,2)	1	1 367	29	0	18 040	6
	2	732	0	0	1 940	3
	3	857	52	2	28 580	8
IA(3,50,1)	1	3 016	20	4	18 790	25
	2	2 262	0	6	7 290	19
	3	2 286	0	6	7 350	28
IA(3,50,3)	1	2 975	8	0	11 680	22
	2	2 354	13	3	13 210	23
	3	2 303	8	1	10 050	28
IA(3,70,1)	1	3 794	768	43	402 590	44
	2	1 728	395	30	208 540	21
	3	4 770	617	51	332 310	68
IA(5,20,1)	1	450	0	6	2 670	<1
	2	806	36	6	21 480	3
	3	480	0	0	1 200	1
	4	930	0	6	3 870	6
	5	446	24	0	13 060	1
IA(5,20,3)	1	456	0	0	1 140	<1
	2	736	0	0	1 940	3
	3	462	0	18	5 700	<2
	4	960	0	0	2 400	5
	5	434	18	0	10 140	<1

Tableau B.1 Résultats de la première phase de l'heuristique en deux phases (suite)

Instance	Id Dep	Demande	U^{finale}	O^{finale}	Objectif	Temps (sec)
IA(5,50,1)	1	1 822	498	25	258 750	10
	2	726	181	14	95 480	4
	3	2 140	446	36	236 490	20
	4	1 692	125	34	75 040	6
	5	1 068	299	21	156 780	6
IA(5,70,1)	1	1 547	267	32	144 940	12
	2	1 121	290	22	152 730	3
	3	4 711	719	69	387 180	68
	4	1 436	301	17	157 840	10
	5	1 349	318	23	167 590	7
IA(5,70,2)	1	3 354	366	94	214 290	12
	2	2 667	384	75	216 655	43
	3	159	4	0	2 395	<1
	4	2 597	169	76	109 800	14
	5	2 161	204	60	122 070	7
IA(5,200,1)	1	3 477	745	86	401 110	10.
	2	8 792	1 190	167	656 225	440
	3	1 704	151	64	95 550	9
	4	6 660	327	227	236 660	50
	5	10 634	756	254	466 855	62
IA(10,200,1)	1	3 582	671	73	361 280	10
	2	4 114	835	77	440 400	414
	3	6 388	415	232	281 145	39
	4	2 064	142	79	95 765	8
	5	2 312	135	59	87 840	10
	6	6 272	237	157	171 730	88
	7	951	189	31	104 250	<1
	8	2 247	153	71	99 695	8
	9	1 600	199	51	115 885	4
	10	1 637	92	43	60 750	7

Tableau B.1 Résultats de la première phase de l'heuristique en deux phases (suite et fin)

Instance	Id Dep	Demande	U^{finale}	O^{finale}	Objectif	Temps (sec)
IA(10,300,1)	1	8 265	1214	174	668 690	26
	2	3 594	1 348	54	680 505	26
	3	5 872	397	221	268 110	34
	4	9 268	492	292	341 725	65
	5	2 336	341	52	188 630	8
	6	1 959	158	76	102 715	9
	7	1 085	312	19	162 745	<1
	8	6 110	431	181	275 425	38
	9	2 227	228	63	134 975	9
	10	5 984	429	138	263 265	45
IA(10,400,1)	1	3 280	1 138	74	593 085	12
	2	1 506	890	15	450 340	<1
	3	13 415	828	427	553 500	94
	4	1 229	47	46	38 080	4
	5	6 298	318	176	218 435	38
	6	6 888	236	170	177 565	57
	7	1 560	498	32	259 750	2
	8	7 181	529	213	334 955	44
	9	7 666	373	242	265 850	49
	10	13 231	673	306	445 200	474

Tableau B.2 Résultats de la première approche de la phase ré-optimisation de l'heuristique en deux phases

Instance	U^{init}	O^{init}	U^{finale}	O^{finale}	Nb Transferts	Objectif	Gap (%)	Temps (sec)
Instances de petite taille								
IA(2,20,1)	16	0	2	2	2	9 520	0	2
IA(2,20,4)	37	9	30	10	1	25 310	120	<1
IA(2,50,1)	1 080	59	916	104	37	503 800	9,3	3
IA(2,50,3)	99	3	99	3	0	69 410	194	<1
IA(3,20,1)	24	0	24	0	0	19 700	105	<1
IA(3,20,2)	72	0	43	11	3	32 110	213	<1
IA(3,50,1)	16	12	12	12	1	28 460	31	<1
IA(3,50,3)	28	4	27	3	3	33 850	59	<1
IA(3,70,1)	1 743	96	1 553	187	63	852 230	6,9	8
IA(5,20,1)	52	12	23	31	5	27 390	143	<1
IA(5,20,3)	18	18	18	18	0	21 320	95	<1
IA(5,50,1)	1 497	101	1 114	161	62	618 940	14	7
Instances de grande taille								
IA(5,70,1)	1 807	121	1 537	193	64	844 860	4,6	15
IA(5,70,2)	1078	276	953	322	19	583954	205	3
IA(5,200,1)	3060	698	2670	820	72	1617669	200	2
IA(10,200,1)	2973	801	2421	961	121	1531650	140	9
IA(10,300,1)	5203	1150	3746	1491	234	2370589	175	32
IA(10,400,1)	5375	1562	4576	1689	161	2866840	170	31

Tableau B.3 Résultats de la deuxième approche de la phase ré-optimisation de l'heuristique en deux phases

Instance	Demande	U^{init}	O^{init}	U^{finale}	O^{finale}	Nb Transferts	Objectif	Gap (%)	Temps (sec)
Instances de petite taille									
IA(2,20,1)	3 140	16	0	2	2	2	9 520	0	1,8
IA(2,20,4)	3 052	37	9	31	3	4	24 360	112	<1
IA(2,50,1)	7 347	1 080	63	888	111	58	495 160	7	20
IA(2,50,3)	7 604	98	2	62	2	6	51 149	116	1,2
IA(3,20,1)	2 968	24	0	12	8	4	15 930	66	<1
IA(3,20,2)	2 956	72	0	32	0	5	24 060	134	<1
IA(3,50,1)	7 564	16	12	12	12	1	28 460	31	<1
IA(3,50,3)	7 632	28	4	26	2	3	33 080	56	1,7
IA(3,70,1)	10 292	1 743	96	1 530	140	106	830 160	4	44
IA(5,20,1)	3 112	30	18	14	10	4	17 470	55	<1
IA(5,20,3)	30 48	20	12	6	6	2	12 400	13	1,1
IA(5,50,1)	7 448	1 502	102	1 043	120	99	576 089	6	59
Instances de grande taille									
IA(5,70,1)	10 164	1 807	125	1 505	151	119	820 649	1	70
IA(5,70,2)	10 938	1 078	276	765	242	111	476 350	149	35
IA(5,200,1)	31 267	3 060	698	2 243	499	423	1 349 070	150	1 861
IA(10,200,1)	31167	2973	801	1889	696	445	1 221 930	91	11 234
IA(10,300,1)	46 700	5 203	1 150	2 252	955	850	1 569 364	82	11 232
IA(10,400,1)	62 254	5 375	1 562	2 995	1 113	968	1 991 589	88	14 300

Tableau B.4 Résultats de la troisième approche de la phase ré-optimisation de l'heuristique en deux phases

Instance	Demande	U^{init}	O^{init}	U^{finale}	O^{finale}	Nb Transferts	Objectif	Gap (%)	Temps (sec)
Instances de petite taille									
IA(2,20,1)	3 140	16	0	2	2	2	9 520	0	2
IA(2,20,4)	3 052	37	9	13	9	5	17 100	49	1
IA(2,50,1)	7 347	1 080	63	870	122	67	487 159	5	13
IA(2,50,3)	7 604	98	2	62	2	6	51 150	116	<1
IA(3,20,1)	2 968	24	0	12	8	4	15 930	66	<1
IA(3,20,2)	2 956	72	0	32	0	5	24 060	134	<1
IA(3,50,1)	7 564	16	12	8	8	1	25 480	17	<1
IA(3,50,3)	7 632	28	4	22	2	7	31 370	48	2
IA(3,70,1)	10 292	1 743	96	1 530	140	91	828 100	3	53
IA(5,20,1)	3 112	30	18	14	10	4	174 70	55	<1
IA(5,20,3)	3 048	20	12	6	6	2	12 400	13	<1
IA(5,50,1)	7 448	1 502	102	1 039	122	99	574 110	5	51
Instances de grande taille									
IA(5,70,1)	10 164	1 807	125	1 504	149	109	818 560	1	79
IA(5,70,2)	10 938	1 078	276	764	185	119	461 760	141	3 649
IA(5,200,1)	31 267	3 060	698	1 578	520	383	1 056 954	96	3 877
IA(10,200,1)	31 167	2 973	801	1 482	741	576	1 036 895	63	23 213

ANNEXE C RÉSULTATS DE L'HEURISTIQUE EN DEUX PHASES AVEC LA NOUVELLE APPROCHE DE LA PREMIÈRE PHASE

Tableau C.1 Résultats de la deuxième phase adaptée à la nouvelle première phase (Temps de calcul Rapide)

Instance	Demande	U^{init}	O^{init}	U^{finale}	O^{finale}	Nb Transferts	Objectif	Gap (%)	Temps (sec)
Instances de petite taille									
IA(2,20,1)	3 140	0	0	0	0	2	8 010	0	1
IA(2,20,4)	3 052	5	4	5	3	2	11 180	38	<1
IA(2,50,1)	7 347	842	99	835	77	38	455 807	3	40
IA(2,50,3)	7 604	29	48	35	27	8	44 040	121	<1
IA(3,20,1)	2 968	0	5	0	5	1	9 087	0	<1
IA(3,20,2)	2 956	2	2	1	2	5	9 007	5	<1
IA(3,50,1)	7 564	8	0	0	0	3	19 580	0	<1
IA(3,50,3)	7 632	8	0	0	0	4	19 670	0	<1
IA(3,70,1)	10 292	1 492	136	1474	98	56	797 180	2,7	63
IA(5,20,1)	3 112	12	0	8	0	4	12 045	49	<1
IA(5,20,3)	3 048	2	2	2	2	2	9 420	5	1
IA(5,50,1)	7 448	1 162	140	1 042	75	84	559 874	3	72
Instances de grande taille									
IA(5,70,1)	10 164	1 500	152	1 491	121	54	800 985	1	107
IA(5,70,2)	10 938	495	439	398	397	83	330 312	72	338
IA(5,200,1)	31 267	820	1 070	769	1 050	196	735 807	36	5
IA(10,300,1)	46 700	5 888	1 400	4 626	741	484	2 652 632	208	633
IA(10,200,1)	31 167	890	1 101	852	1 090	194	786 889	23	9
IA(10,400,1)	62 254	6 282	2 724	5 664	2 307	317	3 580 730	238	25

Tableau C.2 Résultats de la deuxième phase adaptée à la nouvelle première phase (Temps de calcul allongé)

Instance	Demande	U^{init}	O^{init}	U^{finale}	O^{finale}	Nb Transferts	Objectif	Gap (%)	Temps (sec)
Instances de petite taille									
IA(2,20,1)	3 140	0	0	0	0	2	8 010	0	1
IA(2,20,4)	3 052	5	4	5	3	2	11 180	38	<1
IA(2,50,1)	7 347	844	97	830	89	35	456 125	3	40
IA(2,50,3)	7 604	5	0	1	0	10	20 657	3	<1
IA(3,20,1)	2 968	0	5	0	5	1	9 087	0	<1
IA(3,20,2)	2 956	2	2	0	2	5	8 510	0	<1
IA(3,50,1)	7 564	8	0	0	0	3	19 580	0	<1
IA(3,50,3)	7 632	8	0	0	0	4	19 670	0	<1
IA(3,70,1)	10 292	1481	131	1477	97	65	786 934	1,4	124
IA(5,20,1)	3 112	12	0	8	0	4	12 045	49	<1
IA(5,20,3)	3 048	2	2	2	2	2	9 420	5	1
IA(5,50,1)	7 448	1131	97	1 063	72	94	571 079	5	171
Instances de grande taille									
IA(5,50,1)	7 448	1 131	97	1 063	72	94	571 079	5	171
IA(5,70,1)	10 164	1 489	132	1 487	101	81	795 829	0	95